

Prime-Only Summation as a Generator Measure: Euler-Product Thermodynamics, Irreducible Degrees of Freedom, and Testable Scaling Laws

Douglas C. Youvan

doug@youvan.com

www.youvan.ai

February 9, 2026

Prime-restricted summation is usually treated as a technical device of analytic number theory: one sums over primes rather than over all integers, and obtains familiar approximations weighted by $1/\log x$. This paper advances a different claim: prime-only summation can be structurally correct, not merely convenient, whenever the system being modeled has a generator–configuration architecture. In such architectures, primes play the role of irreducible degrees of freedom, while composites represent composed configurations that should not be counted as independent contributors. The consequence is a principled replacement of uniform integer aggregation by a prime-indexed generator measure, with composites recovered by a composition operator rather than summed as primitives. We formalize this viewpoint through additive energies over prime exponents, derive Euler-product partition functions, and show how macroscopic observables become prime-controlled response laws. The framework yields falsifiable signatures: log-log divergences, $N/\log N$ growth in independent mode counts, and distinct cutoff behaviors under spectral weighting. We also propose an information-theoretic novelty functional that avoids double-counting by treating prime factors as basis elements and composites as redundancies. The result is a unified “irreducible aggregation” calculus linking prime sums, factorization thermodynamics, and empirical tests for multiplicative generative structure.

Keywords: prime-only summation, prime-restricted sums, Euler product, partition function, zeta function thermodynamics, irreducible degrees of freedom, generator measure, multiplicative generative models, additive energy over prime exponents, scaling laws, log-log divergence, Chebyshev theta function, prime zeta function, information theory, novelty functional, overcounting correction, compositional state space, sparse sampling, analytic number theory, statistical mechanics analogy.

A collaboration with GPT-5.2-Thinking. 61 pages. CC4.0.

Outline

1. Introduction: From Prime Sums as Technique to Prime Sums as Ontology

- 1.1 Prime-restricted summation in standard analytic practice
- 1.2 The generator–configuration distinction and why it matters
- 1.3 The main thesis: primes as independent degrees of freedom
- 1.4 What would make the thesis nontrivially true and testable
- 1.5 Roadmap of results and predictions

2. Mathematical Preliminaries: Prime-Indexed Aggregation and Composition

- 2.1 Prime-only summation operators and normalization choices
- 2.2 Density heuristics and integral surrogates for prime sums
- 2.3 Factorization as a free commutative monoid and uniqueness
- 2.4 Generator measures versus configuration measures
- 2.5 Avoiding double-counting: composites as emergent states

3. Generator Energy and Additive Cost on Prime Exponents

- 3.1 Defining energies on primes and extending to integers additively
- 3.2 The choice $e(p)=\log p$ and the emergence of zeta partition forms
- 3.3 Alternative spectra: power laws, logs, and empirically fitted energies
- 3.4 Interpretation: single-mode spectra versus multi-occupancy configurations
- 3.5 Identifiability: when can $e(p)$ be learned from data

4. Euler-Product Thermodynamics: Partition Functions from Prime Generators

- 4.1 Partition functions over configurations and factorization into primes
- 4.2 Free energy, expected energy, and response functions as prime sums
- 4.3 Critical behavior near $\beta=1$ and controlled divergences
- 4.4 Cutoffs, regularization, and finite-size analogs
- 4.5 What “phase transitions” mean in a generator–configuration model

5. Prime-Controlled Scaling Laws and Falsifiable Signatures

- 5.1 Independent mode counts: $N/\log N$ as a structural fingerprint
- 5.2 Log-log divergence signatures and what they imply operationally
- 5.3 Comparing prime-aggregation scaling to integer-aggregation scaling
- 5.4 Robustness under reweighting and changes of observational window
- 5.5 Distinguishing multiplicative composition from additive or random models

6. The Irreducible Aggregation Principle: A General Counting Theorem

- 6.1 Formal statement: when prime-only is the correct primitive sum
- 6.2 Reconstruction of configuration totals via composition operators
- 6.3 Relations to Euler products, logarithms of generating functionals

- 6.4 Error terms and what can and cannot be controlled
- 6.5 Conceptual payoff: “primes as basis” without metaphor
- 7. An Information-Theoretic Novelty Functional Built on Primes
 - 7.1 Defining prime-novelty and prime-surprise for observed datasets
 - 7.2 Divisibility statistics and expected prime signatures under mixing
 - 7.3 Overcounting corrections: composites as redundant descriptions
 - 7.4 Practical estimators and finite-sample behavior
 - 7.5 Predictions: how novelty grows under multiplicative generative dynamics
- 8. Model Selection and Diagnostics: Testing for Multiplicative Generative Structure
 - 8.1 Competing hypotheses: additive generation, random labeling, mixed regimes
 - 8.2 Diagnostic plots: scaling curves and response functions
 - 8.3 Sensitivity to noise, missingness, and observational bias
 - 8.4 Null models and permutation baselines
 - 8.5 What would falsify the prime-generator ontology
- 9. Applications and Interpretive Domains
 - 9.1 Physical analogs: spectra, modes, and compositional state spaces
 - 9.2 Informational systems: codes, logs, event streams, and description length
 - 9.3 Networks and systems engineering: independent failure modes and coupling
 - 9.4 Cognitive and semantic systems: irreducible causes versus derived narratives
 - 9.5 Guidance for choosing $e(p)$ and $w(p)$ in applied settings
- 10. Limitations, Open Problems, and Research Program
 - 10.1 Where the analogy breaks and where it becomes mathematics
 - 10.2 Dependence on factorization uniqueness and domain encoding
 - 10.3 Extending beyond primes: irreducibles in other algebraic structures
 - 10.4 Computational questions: estimation, complexity, and scaling in practice
 - 10.5 A research agenda: measurable predictions and experimental designs
- 11. Conclusion: Prime-Only Summation as a Measure of Independent Reality
 - 11.1 Summary of claims and what is genuinely new
 - 11.2 The irreducible aggregation lens as a unifying viewpoint
 - 11.3 Final statement of falsifiable signatures and next steps
- 12. References

Art

1. Introduction: From Prime Sums as Technique to Prime Sums as Ontology

Prime-restricted summation is one of those ideas that feels immediately intuitive once stated: instead of summing a function across all integers, we sum it only across the primes and ignore the rest. In most mathematical contexts, this is introduced as a technical maneuver. Primes are sparse; they carry multiplicative structure; they are where the Euler product lives; and restricting attention to primes often clarifies asymptotics that are washed out when composites are included. Yet that standard framing leaves a deeper possibility mostly unspoken: what if prime-only summation is not merely a useful restriction, but the correct aggregation rule for certain kinds of systems?

This paper argues that such a situation is not only conceivable, but structurally forced whenever the system being modeled has a generator–configuration architecture. In that architecture, primes correspond to irreducible degrees of freedom, while composite integers correspond to composed configurations built from those irreducibles. If this is the right ontology, then summing over all integers is not “more complete” than summing over primes; it is a different operation entirely, one that mixes the primitive objects (generators) with their composites (configurations) and thereby risks double-counting what should not be counted independently.

The introduction sets up this shift in viewpoint. The technical device becomes a structural principle: primes are not just a convenient subset of integers; they are the canonical index set of independent multiplicative generators. The central claim is that prime-only summation becomes the correct primitive measure when the phenomenon under study is compositional in a multiplicative sense. In that case, the full integer sum is still relevant, but it must appear as a derived quantity reconstructed from prime-level data via a composition operator (Euler product, log-generating functional, or related transforms), not as the primitive aggregation itself.

1.1 Prime-restricted summation in standard analytic practice

In standard analytic number theory, sums over primes arise because primes are the “atoms” of multiplication. When you seek asymptotic information about arithmetic functions, it is natural to separate the behavior of primes from the behavior of general integers. Prime-only sums such as the prime-counting function, Chebyshev-type sums, or sums of reciprocal primes appear as foundational objects. One reason is practical: the density of primes near x behaves like $1/\log x$, so prime-only sums often admit integral approximations that differ systematically from integer sums. Another reason is structural: multiplicative phenomena frequently linearize when expressed at the prime level, particularly after taking logarithms of Euler products.

But crucially, in this standard practice, the prime-only sum is still treated as one statistic among many. It is a lens, not a law. If one wants a “complete” count or “complete” sum, the default

instinct is to sum over all integers. Prime-only summation is then framed as sparse sampling: informative, sometimes elegant, but inherently partial.

This paper flips the polarity. We keep all of the classic motivations, but we reinterpret the role. Instead of prime-only sums being “partial views” of integer sums, we propose conditions under which prime-only sums are the correct primitive aggregation and integer sums are the derived views.

1.2 The generator–configuration distinction and why it matters

Many scientific and informational systems have an architecture that can be described abstractly as generator–configuration.

A generator is an irreducible building block: it cannot be decomposed into smaller independent pieces within the chosen ontology. A configuration is a composed object: it is built from generators, often with multiplicities, interactions, or couplings. In a physical theory, generators might correspond to fundamental modes, basis excitations, or independent constraints, while configurations correspond to multi-mode states. In an informational theory, generators might correspond to primitive symbols or independent causes, while configurations correspond to compositions, mixtures, or redundancies. In a network or engineering context, generators might be independent failure modes or root causes, while configurations are cascading combinations.

The distinction matters because aggregation depends on what you consider independent. If you want to measure the number of independent degrees of freedom, you do not count every configuration as an additional degree of freedom. You count the generators. If you want to measure the distribution of realized configurations, you count configurations, but you do so knowing that the configuration counts are not independent facts; they are induced by generator statistics plus a composition rule.

In arithmetic, the primes are the generators of the multiplicative monoid of integers, and composites are configurations. That statement is not poetic; it is exact. Every positive integer is a unique product of primes with integer exponents. This gives a canonical decomposition into generators and occupancies. It follows that any model in which the integer label n represents a composed state in a multiplicative composition system has, baked into it, an implied generator–configuration architecture. And in that setting, prime-only summation is no longer a cute restriction; it is the natural sum over generators.

1.3 The main thesis: primes as independent degrees of freedom

The central thesis can be stated in one line:

Prime-only summation is the correct primitive aggregation whenever the phenomenon indexed by integers is generated by multiplicative composition of irreducible modes, and composites therefore represent configurations rather than independent primitives.

In such systems, primes correspond to independent degrees of freedom, and the exponent vector of prime occupancies is the true configuration coordinate. Summing over all integers is then equivalent to summing over all occupancy configurations. That may be exactly what you want in some contexts (for example, a partition function over configurations), but it is not the correct way to count or aggregate independent contributions. The correct primitive aggregation is over the generators themselves.

The thesis therefore separates two questions that are often conflated:

What are the independent contributors? This is prime-level.

What are the realized composed states? This is integer-level and reconstructible.

Once separated, the intellectual move is simple but powerful: if you build a model of a system and your observables empirically behave as though the relevant degrees of freedom grow like the number of primes up to a cutoff, then you have evidence that the system's natural basis is prime-like: sparse, generator-indexed, and compositionally reconstructible.

This paper does not claim that the world is literally made of primes. It claims something more precise and testable: that certain datasets and phenomena are better modeled by an irreducible generator measure than by a uniform configuration measure, and that primes provide the canonical example of such a generator index set whenever multiplicative composition is the operative mechanism.

1.4 What would make the thesis nontrivially true and testable

To avoid collapsing into analogy, the thesis must cash out as predictions that differ from what you would obtain by summing over all integers, and those predictions must be empirically checkable within the chosen domain.

The thesis becomes nontrivially true if all four of the following hold:

First, the domain has a compositional encoding where integer labels correspond to multiplicative composition of irreducibles. This can be literal (a system defined by multiplication and factorization) or representational (a mapping from domain objects to integers where composition corresponds to multiplication).

Second, there exists a principled reason not to treat composites as independent. This typically means that composites are redundant once generators are known, or that counting them as independent double-counts novelty, capacity, or causality.

Third, generator-level aggregation yields quantitative scaling laws that are distinct from configuration-level aggregation. The classic signatures include sparse growth rates (for example, $N/\log N$ rather than N) or slow divergences (log log behavior) in appropriately weighted observables. The point is not to worship particular functions; the point is that prime-density effects produce characteristic scaling that does not appear under uniform integer aggregation unless it is artificially imposed.

Fourth, the mapping from prime-level observables to configuration-level behavior is explicit. That is, one can show a reconstruction rule: the full configuration statistics are derived from prime-level spectra by a composition operator, often factorized. This gives the thesis operational bite: you do not merely say “primes are fundamental,” you show that they are sufficient to generate the higher-level observables and that treating composites as primitives degrades interpretability or prediction.

When these conditions are satisfied, prime-only summation is no longer a stylistic choice. It becomes a measurement choice: you are measuring generators rather than configurations.

1.5 Roadmap of results and predictions

The remainder of the paper develops the thesis in a sequence of steps designed to separate conceptual clarity from technical machinery and to produce falsifiable signatures.

We first formalize prime-only summation as a generator-indexed measure and distinguish it from configuration-indexed summation. We then introduce additive energies or costs defined on primes and extended to integers via prime-exponent additivity. This naturally yields factorized generating objects: Euler-product partition functions, log-generating relations, and prime-controlled response functions.

From that framework we derive predictions of three kinds.

The first are scaling predictions: the growth of independent mode counts, the behavior of weighted sums under cutoffs, and the appearance of slow divergences in specific regimes. These are signatures that, if absent, argue against a prime-generator ontology for the domain.

The second are reconstruction predictions: configuration-level distributions should be recoverable from prime-level spectra by explicit composition operators. When this fails, it indicates that the domain’s composition rule is not multiplicative or that the proposed encoding is wrong.

The third are diagnostic predictions for model selection: prime-based novelty measures should distinguish multiplicative generative structure from additive generation, random labeling, or mixed regimes. This creates an empirical workflow: do not argue about primes philosophically; test whether prime-controlled laws actually describe the data better than uniform aggregation.

By the end, the claim is not that primes are magical, but that irreducible aggregation is a principled alternative to naive counting, and that primes provide a uniquely rigorous prototype of how a generator measure can replace a configuration measure without losing completeness. In the sections that follow, we make that prototype explicit, and we show exactly what would count as evidence for or against it.

2. Mathematical Preliminaries: Prime-Indexed Aggregation and Composition

This section establishes the mathematical scaffolding for treating prime-only summation as a generator-indexed aggregation rule rather than a mere restriction of an integer-indexed sum. The aim is not to re-derive classical number theory, but to isolate a small set of definitions and distinctions that become decisive once we interpret primes as independent degrees of freedom and composites as configurations. In particular, we will distinguish between measures over generators and measures over configurations, show how composition reconstructs configuration totals from generator data, and clarify where overcounting enters if composites are treated as primitives.

2.1 Prime-only summation operators and normalization choices

Let f be a real- or complex-valued function on the positive integers (or on primes). A standard “prime-only” finite sum is written as a restriction of the index set:

$$S_p(N; f) = \sum_{\text{primes } p \leq N} f(p).$$

We will treat this as the basic generator-indexed aggregation operator. In contrast, the usual integer-indexed aggregation is:

$$S(N; f) = \sum_{\text{integers } 1 \leq n \leq N} f(n).$$

The distinction is obvious at the level of notation but becomes profound once f is interpreted as an observable associated either to generators (primes) or to configurations (all integers). In the generator–configuration worldview, it is not meaningful to compare these sums as though one were “partial” and the other “complete.” They average over different objects.

Because primes are sparse, many prime-indexed sums require normalization if one wants a stable notion of average. Three normalizations are common and will be useful depending on the interpretive goal.

First, the unnormalized prime sum, which is appropriate when the sum itself is the observable.

Second, the prime-average up to N :

$$A_p(N; f) = (1 / \pi(N)) \sum_{p \leq N} f(p),$$

where $\pi(N)$ is the number of primes $\leq N$. This corresponds to an arithmetic mean over generators up to a cutoff.

Third, the log-weighted prime average:

$$L_p(N; f) = (1 / \sum_{p \leq N} \log p) \sum_{p \leq N} f(p) \log p.$$

This is not an aesthetic trick. The log weights appear naturally when one wants a generator measure that matches multiplicative composition. The quantity $\sum_{p \leq N} \log p$ (the Chebyshev theta function) behaves like N in a precise asymptotic sense, which means log-weighted prime sums can behave like “uniform” integrals over a continuum scale. In generator-configuration terms, log weighting often corresponds to measuring primes in proportion to their multiplicative “size” rather than their count.

Normalization is therefore not a technical afterthought. It encodes what you mean by a “typical generator”: typical by count, or typical by multiplicative scale, or typical by some domain-specific weight.

2.2 Density heuristics and integral surrogates for prime sums

A prime-only sum is rarely tractable in closed form for general f , but there is a widely used heuristic that replaces discrete prime sampling by a weighted integral based on prime density. Informally, primes near x occur with density about $1/\log x$. This suggests the surrogate:

$$\sum_{p \leq N} f(p) \text{ is approximately } \int_2^N f(x) / \log x \, dx.$$

This approximation is not merely a computational convenience; it shows that prime-only aggregation behaves like integration against a non-uniform measure. In other words, “sum over primes” resembles “integrate with a slowly varying thinning factor.”

This is the first place where the ontology shift starts to appear: if primes are generators, then the generator measure is not uniform on the scale of x . The measure has a natural weight determined by generator sparsity. If your domain exhibits a $1/\log$ scaling in the density of independent modes, you should not normalize it away as a nuisance. That $1/\log$ factor is the signature of prime-like generator structure.

Two consequences matter for later sections.

First, prime-only aggregation is naturally sensitive to slowly varying functions. For example, the difference between $f(x)$ and $f(x)/\log x$ often dominates asymptotics. This gives prime-only observables their distinctive scaling laws.

Second, there is a conceptual separation between generator density and generator value. The $1/\log x$ factor belongs to the generator “counting measure,” not to f itself. When modeling, you must decide whether f already includes an intrinsic weighting, or whether weighting belongs to the measure. Confusing these two is one way systems get misinterpreted as having unusual scaling when in fact only the measure has changed.

2.3 Factorization as a free commutative monoid and uniqueness

The formal reason primes qualify as generators is the unique factorization of integers. Every integer $n \geq 2$ can be written uniquely as

$$n = \text{product over primes } p \text{ of } p^{a_p},$$

where the exponents a_p are nonnegative integers and only finitely many are nonzero.

This implies that the set of positive integers under multiplication is a free commutative monoid generated by primes. “Free” means there are no relations among generators other than commutativity; “commutative” means order does not matter; “monoid” means there is an identity element 1 and associative multiplication.

What matters for this paper is not the abstract algebra for its own sake, but the interpretive consequence: every configuration n corresponds to a unique occupancy vector (a_p) . The occupancy vector is the true coordinate of the configuration in generator space.

Thus, when you sum over all integers, you are summing over all occupancy vectors. When you sum over primes, you are summing over basis directions in that occupancy space. The move from primes to integers is a move from generators to configurations.

This is the mathematical skeleton beneath the structural claim: composites do not add new independent coordinates. They are points in the occupancy lattice.

2.4 Generator measures versus configuration measures

We now formalize the key distinction: a measure over generators and a measure over configurations are not interchangeable.

A generator measure assigns weight to primes. For example, a generator-measure expectation of a prime observable $g(p)$ could take the form

$E_{\text{gen}}[g] = \text{sum over primes } p \text{ of } g(p) * w(p),$

for some weight $w(p)$ determined by the application. If $w(p)=1$, this is a simple counting measure over generators. If $w(p)=\log p$, it is a size-weighted generator measure. If $w(p)=p^{(-\beta)}$, it becomes a spectral or thermal weight, emphasizing smaller primes more strongly.

A configuration measure assigns weight to integers n . For example, a configuration expectation of an integer observable $F(n)$ might be

$E_{\text{conf}}[F] = \text{sum over } n \geq 1 \text{ of } F(n) * W(n),$

with a configuration weight $W(n)$ that might encode a cutoff, a probability distribution, or a Boltzmann factor.

The important point is that these measures answer different questions.

Generator measures are appropriate when you care about independent degrees of freedom: which irreducibles exist, what their spectrum is, and how their contributions scale with cutoff.

Configuration measures are appropriate when you care about realized composed states: how frequently certain combinations occur, what the total partition function is, or what distribution of multiplicative compositions you observe.

In a generator–configuration architecture, configuration weights are often induced from generator weights by a composition rule. For example, if each generator p can appear with multiplicity a_p , and if those multiplicities are independently weighted, the configuration weight factorizes across primes. This is exactly the structure that yields Euler products in later sections.

Thus, in the structural view, a generator measure is fundamental and a configuration measure is derived, unless the domain itself privileges configurations as primary objects.

2.5 Avoiding double-counting: composites as emergent states

We can now state the overcounting problem precisely.

If composites are treated as independent primitives, then aggregation over all integers risks counting the same underlying generator information multiple times. This is not a philosophical complaint; it is a specific structural redundancy.

Consider an observable intended to measure “novelty,” “independent capacity,” “independent causes,” or “basis complexity.” If you sum such an observable over all integers, then every composite n includes prime factors that have already appeared in other integers. Unless your observable is specifically defined to remove redundancy, the integer sum will repeatedly re-count generator contributions.

The prime-only sum, by contrast, counts each generator once by construction.

There are two standard ways to see the double-counting mechanism.

First, through occupancy representation. Each integer n corresponds to exponents (a_p) . A composite corresponds to multiple nonzero exponents or larger exponents. If you count n as a primitive object with its own full independent contribution, you are counting a configuration as though it introduced new generator axes, which it does not. You are counting points in the lattice as if they were basis vectors.

Second, through logarithms of Euler products. Many global configuration sums factorize into products over primes. Taking a logarithm transforms multiplicative factorization into an additive decomposition across primes, revealing that the “independent” content of the configuration sum is prime-indexed. In this view, the integer-level partition function is a composed object, while its log is a generator-level object. Treating integer configurations as primitives is precisely what hides the generator decomposition and invites interpretive overreach.

The practical upshot is an aggregation principle:

If the target quantity is fundamentally about independent degrees of freedom, causes, or generators, aggregate over primes (or irreducibles). If the target quantity is about the full ensemble of composed configurations, aggregate over integers, but interpret the result as a derived composite measure and analyze it via its prime factorization.

This section has been intentionally abstract. It supplies the definitions needed to make the rest of the paper concrete. In the next section we introduce additive energies on primes and extend them to integers through prime exponents. That move will produce explicit Euler-product partition functions and turn the conceptual generator–configuration distinction into an operational calculus with measurable predictions.

3. Generator Energy and Additive Cost on Prime Exponents

If prime-only summation is to function as more than a restatement of “primes are important,” it must be coupled to a principled mechanism by which prime-level information generates integer-level behavior. The cleanest mechanism is additive cost on prime exponents: assign each prime p a generator energy (or cost, or description length) $e(p)$, and extend that assignment to every integer n by declaring that the cost of a configuration is the sum of the costs of the generators used to build it, counted with multiplicity. This is exactly the structure that makes Euler products inevitable rather than optional. It also yields testable predictions: once you choose $e(p)$, you can compute a full family of configuration-level quantities and check whether observed data follows the implied scaling and response laws.

This section builds that mechanism carefully. We define generator energy on primes, extend it additively to all integers using prime exponents, show why the specific choice $e(p)=\log p$ produces zeta-like partition functions, and then broaden to alternative spectra that may be more appropriate for physical or informational systems. Finally, we address identifiability: under what conditions can one infer the generator spectrum $e(p)$ from data, and when is such inference provably underdetermined.

3.1 Defining energies on primes and extending to integers additively

Let P denote the set of primes. Define a function

$$e: P \rightarrow \mathbb{R}_+,$$

interpreted as a nonnegative energy, cost, or penalty assigned to each prime generator. We then extend e to a function on all integers $n \geq 1$ by using the unique factorization of n . Write

$$n = \text{product over primes } p \text{ of } p^{a_p(n)},$$

where $a_p(n)$ is the exponent of p in the prime factorization of n (with $a_p(n)=0$ for all but finitely many primes). Define the configuration energy

$$E(n) = \text{sum over primes } p \text{ of } a_p(n) * e(p),$$

with the convention $E(1)=0$.

This definition imposes a crucial property: E is additive under multiplication,

$$E(mn) = E(m) + E(n),$$

whenever m and n are integers. In other words, multiplication of integers corresponds to composition of configurations, and energies add under composition. This is precisely the generator–configuration architecture stated in mathematical terms.

Two immediate clarifications matter.

First, E does not need to coincide with any standard arithmetic function. It is a model-defined cost. The only structural requirement is additivity over prime exponents.

Second, the choice of $e(p)$ determines the “geometry” of configuration space. Different generator spectra weight different primes differently, producing different distributions of configurations that dominate various sums or ensembles. In a physical analogy, $e(p)$ is the one-particle energy spectrum; in an informational analogy, it is the marginal description cost of introducing a new irreducible component.

Once $E(n)$ is defined, one can form configuration ensembles by weighting each configuration by a decreasing function of $E(n)$. The most common is the exponential (Boltzmann) weight:

$$W_{\beta}(n) = \exp(-\beta * E(n)),$$

where $\beta > 0$ is an inverse temperature parameter controlling how strongly cost suppresses large-energy configurations.

At this point, we have a full model class: choose $e(p)$, define $E(n)$, then define any aggregate of interest as a sum over n of a function of $E(n)$. Crucially, this is no longer an arbitrary integer sum; it is a configuration sum induced by a generator spectrum.

3.2 The choice $e(p)=\log p$ and the emergence of zeta partition forms

One generator spectrum stands out because it makes $E(n)$ coincide with a familiar quantity. Let $e(p) = \log p$.

Then

$$E(n) = \sum_p a_p(n) \log p = \log(\prod_p p^{a_p(n)}) = \log n.$$

So with this choice, configuration energy is exactly the logarithm of n . The Boltzmann weight becomes

$$\exp(-\beta * E(n)) = \exp(-\beta \log n) = n^{-\beta}.$$

A configuration sum over all integers weighted by this energy is then

$$Z(\beta) = \sum_{n \geq 1} \exp(-\beta E(n)) = \sum_{n \geq 1} n^{-\beta},$$

which is the classical zeta series form for $\beta > 1$.

But the deeper point is not the appearance of a familiar function. The deeper point is structural inevitability: because E is additive over prime exponents, the configuration sum factorizes into a product over primes. Concretely, write $Z(\beta)$ in terms of prime occupancies:

$$Z(\beta) = \sum \text{over all exponent vectors } (a_p \geq 0, \text{ finitely supported}) \text{ of } \exp(-\beta * \sum_p a_p e(p)).$$

When the exponent variables separate, the sum over configurations factorizes into independent geometric series for each prime:

$$\begin{aligned} Z(\beta) &= \text{product over primes } p \text{ of } (\sum_{a=0 \text{ to infinity}} \exp(-\beta * a e(p))) \\ &= \text{product}_p 1 / (1 - \exp(-\beta e(p))). \end{aligned}$$

For $e(p)=\log p$, $\exp(-\beta e(p))=p^{-\beta}$, so

$$Z(\beta) = \text{product}_p 1 / (1 - p^{-\beta}),$$

the Euler product representation.

This is the generator–configuration thesis made operational. The prime spectrum determines the full configuration partition function. The primes are the independent modes; the integers are their occupancy configurations.

Two consequences become prediction engines.

First, the partition function diverges at a critical point. In this specific case, the zeta series diverges at $\beta=1$, indicating a sharp boundary between parameter regimes where the configuration ensemble is normalizable and where it is not. If a domain is meaningfully modeled by such an induced ensemble, the presence of a threshold-like behavior near some critical β is not an analogy but an expected structural feature.

Second, macroscopic observables become prime sums. For example, the expected occupancy of prime p in the ensemble is

$$E_{\beta}[a_p] = \sum_{a \geq 0} a \cdot q^a / \sum_{a \geq 0} q^a, \text{ with } q = \exp(-\beta e(p)),$$

which evaluates to

$$E_{\beta}[a_p] = q / (1 - q) = 1 / (\exp(\beta e(p)) - 1).$$

Thus the expected generator occupancies are determined directly by $e(p)$. This is a concrete link between a generator spectrum and measurable configuration statistics.

3.3 Alternative spectra: power laws, logs, and empirically fitted energies

The choice $e(p)=\log p$ is mathematically special because it turns configuration energy into $\log n$. But for modeling real systems, the most “natural” spectrum may differ. The point of the framework is that it supports a family of spectra, each producing distinct scaling signatures.

Several spectrum families are especially relevant.

A log-power spectrum:

$$e(p) = (\log p)^\alpha, \alpha > 0.$$

This makes larger primes increasingly expensive in a controlled way. It retains slow variation but reshapes the relative weighting of mid-range vs large primes. Such spectra can model systems where introducing larger-scale generators is disproportionately costly.

A power spectrum:

$$e(p) = p^\gamma, \gamma > 0.$$

This makes large primes extremely expensive. It generates ensembles dominated by small primes and yields strong suppression of configurations involving large generators. In physical

terms, it resembles a rapidly increasing energy ladder; in informational terms, it corresponds to a model where large irreducible components are practically inaccessible.

A mixed spectrum:

$$e(p) = a \log p + b p^\gamma,$$

which can represent a baseline multiplicative size cost plus a superlinear penalty for large generators.

A learned or fitted spectrum:

$e(p)$ is not chosen from a closed form but inferred from data by matching observed generator usage frequencies or configuration statistics. In such settings, $e(p)$ plays the role of a latent spectrum that best explains how configurations are composed.

It is important that different $e(p)$ lead to different prime occupancy laws:

$$E_{\text{beta}}[a_p] = 1 / (\exp(\text{beta } e(p)) - 1).$$

Thus if one can observe effective occupancies or generator usage rates, one can test whether they align with a hypothesized spectrum. This is one of the primary “nontrivial” paths to evidence: the spectrum is not asserted, it is constrained by observed usage.

Finally, note that the partition function under general $e(p)$ remains

$$Z(\text{beta}) = \prod_p 1 / (1 - \exp(-\text{beta } e(p))),$$

whenever occupancies are allowed to be any nonnegative integer. This universal Euler-product form is the mathematical hallmark of generator independence. If the observed system exhibits correlations or constraints among generators, the factorization will break, and that breakage itself becomes diagnostic.

3.4 Interpretation: single-mode spectra versus multi-occupancy configurations

The generator spectrum $e(p)$ is the single-mode information. It tells you the “cost” of activating generator p once.

But configurations allow multiplicity: p can appear with exponent 2, 3, and so on. In physical language, this is a multi-occupancy system. In informational language, it is repeated use of the same irreducible component.

This distinction matters for how you interpret composite integers. A composite can arise in at least two qualitatively different ways:

First, as a multi-generator configuration, such as pq , where two distinct generators p and q each appear once.

Second, as a high-occupancy configuration in one generator, such as p^k , where a single generator appears multiple times.

These are different types of emergence. The first is combinatorial composition across independent generators. The second is repeated amplification of a single generator.

A prime-only aggregation, by itself, distinguishes neither. It simply enumerates the available generators. The additive-energy extension does distinguish them: p^k has energy $k e(p)$, whereas pq has energy $e(p)+e(q)$. Under a given β , these two types of composites will have different probabilities or weights. Thus the framework predicts not only global scaling but the relative prevalence of “many-small” vs “few-large” composition patterns, depending on the curvature of $e(p)$.

This is precisely where the approach ceases to be a slogan and becomes a model. You can ask detailed questions:

Does the system tend to build complexity by combining distinct generators or by repeating a small set of generators?

Does it exhibit a preference for small primes (few fundamental modes) or does it activate larger primes in a way consistent with a particular $e(p)$?

Are observed configurations compatible with independent generator occupancies, or do they require coupling constraints?

The answers to these questions live at the prime-spectrum level and can be compared to data.

3.5 Identifiability: when can $e(p)$ be learned from data

A central practical question is whether the generator spectrum $e(p)$ is identifiable from observations. The answer depends on what is observed and how the domain is encoded.

There are three progressively stronger observational regimes.

First, configuration-only observation. You observe integers n (configurations) but not their prime exponents explicitly. In this regime, $e(p)$ is generally not identifiable without additional assumptions. Many different spectra can produce similar configuration distributions after marginalization. Identifiability requires structure: a parametric family for $e(p)$, restrictions on β , or multiple datasets collected at different effective β s.

Second, factorization-aware observation. You observe n and can compute or estimate the exponents $a_p(n)$. In this regime, identifiability improves because generator usage is directly

observable. If you can estimate the expected occupancy of each prime p under the generating process, you can invert the occupancy law. In the independent-occupancy model, expected occupancy has the form

$$E_{\text{beta}[a_p]} = 1 / (\exp(\text{beta } e(p)) - 1).$$

If beta is known, this relationship determines $e(p)$ up to estimation noise. If beta is unknown, $e(p)$ is identifiable only up to an overall scaling (since multiplying beta by c and dividing $e(p)$ by c preserves $\text{beta } e(p)$). In many applications, that is acceptable: relative energies are what matter.

Third, multi-condition observation. You observe generator occupancies across multiple regimes (different betas , different cutoffs, different experimental conditions). In this regime, identifiability becomes strong. One can fit $e(p)$ by requiring the same spectrum to explain occupancy patterns under multiple conditions. This is where the model becomes falsifiable: if no single spectrum $e(p)$ explains the data across conditions, generator independence or the assumed composition rule fails.

There is also a conceptual identifiability condition: the encoding of the domain into integers must respect multiplicative composition in a meaningful way. If the mapping is arbitrary, primes lose semantic status and become artifacts of labeling. In such a case, any inferred $e(p)$ is a description of the labeling scheme, not of the system. Therefore, the framework requires that the integer representation be justified by domain structure, not chosen for convenience.

In summary, the generator-energy framework provides a disciplined way to move between prime-only aggregation and full configuration ensembles. The choice of $e(p)$ is not a decoration; it is the model's defining structure, and it yields testable predictions about occupancy, scaling, and critical behavior. The next section builds directly on this by developing Euler-product thermodynamics: how to compute macroscopic quantities and how to recognize prime-controlled signatures in data.

4. Euler-Product Thermodynamics: Partition Functions from Prime Generators

The generator-energy construction of the previous section becomes analytically powerful when it is used to build partition functions. The term “thermodynamics” is used here in a precise, operational sense: we define a weighted ensemble over configurations, compute aggregate quantities such as free energy and expected cost, and study how these aggregates change as a control parameter varies. The novelty is not the borrowing of physical language, but the fact that the entire ensemble factorizes across primes whenever generators are independent and costs add on prime exponents. In that case, the partition function is an Euler product, and the

macroscopic response laws are prime sums. This turns the generator–configuration thesis into a working calculus that produces quantitative predictions and clear failure modes.

4.1 Partition functions over configurations and factorization into primes

Start with a generator spectrum $e(p)$ defined on primes and extend it additively to all integers via

$$E(n) = \sum_p a_p(n) e(p),$$

where $a_p(n)$ is the exponent of prime p in n . Consider the canonical configuration ensemble with inverse temperature $\beta > 0$:

$$Z(\beta) = \sum_{n \geq 1} \exp(-\beta E(n)).$$

This sum is over configurations (integers), but because E is additive over prime exponents, Z can be rewritten as a sum over occupancy vectors (a_p) , where each a_p ranges over nonnegative integers and only finitely many are nonzero. The weight factorizes:

$$\exp(-\beta E(n)) = \exp(-\beta \sum_p a_p e(p)) = \prod_p \exp(-\beta a_p e(p)).$$

Thus

$$Z(\beta) = \sum_{\{a_p\}} \prod_p \exp(-\beta a_p e(p)).$$

When the sum is over independent occupancies for each prime, it factorizes into a product of geometric series:

$$\begin{aligned} Z(\beta) &= \prod_p \left(\sum_{a=0}^{\infty} \exp(-\beta a e(p)) \right) \\ &= \prod_p \frac{1}{1 - \exp(-\beta e(p))}. \end{aligned}$$

This is the Euler-product partition function. It encodes a strong structural statement: the configuration ensemble is built from independent generator modes, each mode having an occupancy distribution controlled by $e(p)$ and β .

Several immediate remarks are important.

First, factorization is not a number-theory trick; it is an independence statement. If you later introduce interactions or constraints that couple primes (for instance, forbidding certain combinations, imposing conservation laws, or enforcing correlations), the product form breaks. Thus the Euler product is a diagnostic: it is equivalent to a non-interacting prime-generator model.

Second, the product converges only under conditions that depend on $e(p)$ and β . Those convergence conditions become the analog of thermodynamic stability.

Third, the Euler-product form offers a clean conceptual division: generator physics lives in the factors, configuration physics lives in the product. This makes it natural to compute macroscopic quantities by differentiating $\log Z$, which is a sum over primes rather than a product.

4.2 Free energy, expected energy, and response functions as prime sums

Define the free energy (up to conventions) as

$$F(\beta) = -(1/\beta) \log Z(\beta).$$

The crucial observation is that $\log Z$ linearizes the Euler product:

$$\log Z(\beta) = - \sum_p \log(1 - \exp(-\beta e(p))).$$

Thus any macroscopic quantity derived from $\log Z$ becomes a prime-indexed sum.

The expected configuration energy (expected cost) is given by

$$U(\beta) = E_{\beta}[E] = - d/d \beta \log Z(\beta).$$

Differentiate:

$$d/d \beta \log Z(\beta) = d/d \beta [- \sum_p \log(1 - \exp(-\beta e(p)))].$$

For a single prime p , the derivative gives a standard occupancy formula. The result is

$$\begin{aligned} U(\beta) &= \sum_p e(p) * \exp(-\beta e(p)) / (1 - \exp(-\beta e(p))) \\ &= \sum_p e(p) / (\exp(\beta e(p)) - 1). \end{aligned}$$

This is one of the central “prime-controlled response laws.” It expresses the macroscopic expected energy as a sum of contributions from each generator mode, with a universal occupancy factor.

Similarly, the expected occupancy of a specific prime generator p is

$$N_p(\beta) = E_{\beta}[a_p] = 1 / (\exp(\beta e(p)) - 1).$$

Thus the ensemble decomposes into independent modes, each with a geometric occupancy distribution. The average occupancy is large when $\beta e(p)$ is small, and exponentially small when $\beta e(p)$ is large. This gives an operational meaning to $e(p)$: it is the control knob that decides which primes participate in typical configurations at a given β .

One can also define a “heat capacity” analog, measuring sensitivity of energy to β :

$$C(\beta) = dU/dT \text{ with } T = 1/\beta,$$

or equivalently, up to sign conventions,

$dU/d\beta = -\text{variance}_\beta(E)$.

The key point is that these response functions become prime sums of simple functions of β $e(p)$. This yields a concrete test strategy: if the system truly has independent prime generators with additive costs, then measured response curves should match these prime-sum forms for some spectrum $e(p)$, at least over a stable regime of β .

4.3 Critical behavior near $\beta=1$ and controlled divergences

The most famous special case is $e(p)=\log p$, where $E(n)=\log n$ and the partition function becomes the zeta series

$$Z(\beta) = \sum_{n \geq 1} n^{-\beta},$$

which converges for $\beta > 1$ and diverges for $\beta \leq 1$. In Euler-product form,

$$Z(\beta) = \prod_p 1/(1 - p^{-\beta}).$$

This identifies $\beta=1$ as a critical boundary between normalizable and non-normalizable ensembles. Interpreted in the generator–configuration model, β controls how aggressively large configurations are suppressed. When $\beta > 1$, suppression is strong enough to keep total weight finite; when β approaches 1 from above, the ensemble becomes heavy-tailed and normalization fails at $\beta=1$.

Even more important than the divergence of Z itself is the behavior of derived quantities as β approaches the critical point. Because $\log Z$ is a sum over primes, divergences often appear as slowly growing prime sums. For example, when β is close to 1, the contribution of small primes dominates, and the system can exhibit controlled but unbounded growth in macroscopic observables.

The phrase “controlled divergences” means two things here.

First, the divergence occurs at a specific parameter boundary and has a characteristic asymptotic form. This makes it a diagnostic signature rather than a vague warning that something “blows up.”

Second, the divergence often lives in one macroscopic quantity while others remain finite, depending on how they weight generator contributions. This separation is useful: it allows one to design observables that are sensitive to generator architecture without requiring the system to sit exactly at criticality.

In applications, one need not insist that $\beta=1$ is literally special. The more general claim is: for a given spectrum $e(p)$, there is typically a boundary in β separating regimes where Z converges and diverges, and near that boundary the ensemble shows sharp changes in which

generators are active and how configuration complexity distributes. If your domain exhibits abrupt regime shifts that can be modeled as a change in how generator costs suppress complexity, this framework provides a precise way to encode and test that idea.

4.4 Cutoffs, regularization, and finite-size analogs

Real systems do not sum over all primes or all integers. They have cutoffs: finite resolution, finite resources, finite observation windows, finite maximum complexity. Therefore, the infinite products and infinite sums should be treated as ideal limits, with practical work done using regularized versions.

There are two natural cutoffs.

A prime cutoff $P_{\max}$, where only primes $p \leq P_{\max}$ are allowed as generators. This corresponds to a finite generator set. The partition function becomes

$$Z_{\{P_{\max}\}}(\beta) = \text{product}_{\{p \leq P_{\max}\}} 1/(1 - \exp(-\beta e(p))).$$

An integer cutoff $N_{\max}$, where only configurations $n \leq N_{\max}$ are included:

$$Z_{\{N_{\max}\}}(\beta) = \text{sum}_{\{n \leq N_{\max}\}} \exp(-\beta E(n)).$$

These cutoffs are not equivalent. A prime cutoff limits which generators exist; an integer cutoff limits which configurations are observed. In many applications, both are present implicitly, and distinguishing them clarifies what is being measured.

Cutoffs also allow one to define finite-size analogs of critical behavior. In the infinite system, a true divergence may occur at a critical β . In the finite system, divergences are replaced by rapid crossovers: quantities become large but finite, and their maxima or steepest slopes shift with cutoff size. These finite-size shifts are themselves informative. If the system is genuinely governed by a prime-generator ensemble, then as the cutoff grows, the crossover behavior should move and sharpen in predictable ways. If it does not, the prime-generator hypothesis is weakened.

Regularization can also be done by modifying the spectrum or occupancy rules. For example, one can impose a maximum occupancy per prime, disallowing arbitrarily large exponents. This replaces each geometric series factor with a finite polynomial factor. Such modifications correspond to physical constraints (finite occupancy) or informational constraints (limited repetition) and again provide a clean way to test whether the unconstrained model is appropriate.

The practical moral is that prime-generator thermodynamics is not tied to ideal infinities. The Euler-product structure already exists at finite cutoffs and can be compared directly to empirical data without appealing to asymptotic limits.

4.5 What “phase transitions” mean in a generator–configuration model

The term “phase transition” is often abused when imported into non-physical domains. Here it is used in a restricted and operational sense: a phase transition is a sharp change in the typical structure of configurations as a control parameter varies, driven by a structural boundary in the normalization or moments of the ensemble.

In a generator–configuration model, such transitions have a clear interpretation: they mark a regime change in which sets of generators “turn on” or “turn off” in typical configurations.

When β is large, $\exp(-\beta e(p))$ is small for all but the lowest-cost primes, so typical configurations use only a small subset of generators and have low occupancy. When β decreases, more primes become energetically accessible, and configurations become richer. Near a critical boundary, the ensemble may become dominated by large configurations with many activated generators or high occupancies. Beyond the boundary, the partition function or key moments may fail to exist, meaning the model no longer defines a stable probability distribution without additional constraints.

Thus “phase transition” in this setting does not require literal matter or literal temperature. It requires only three ingredients:

A generator spectrum $e(p)$, an occupancy rule, and a control parameter β that regulates composition complexity.

The payoff is interpretive discipline. If you claim a “transition,” you must specify which quantity ceases to be finite, or which response function becomes singular, or which crossover sharpens with increasing cutoff. That is a testable statement. It avoids the vagueness of metaphor and ties the language to concrete mathematics.

In the context of this paper’s thesis, the main point is that prime-only summation is not an isolated operation. It is the primitive description of generator structure. Euler-product thermodynamics then explains how full configuration behavior emerges from that generator structure, how macroscopic laws become prime sums, and how sharp regime boundaries can occur in a compositional system without any need to treat composites as independent primitives. The next section will exploit this machinery to derive explicit scaling laws and falsifiable signatures that distinguish prime-generator systems from systems whose complexity is not built by multiplicative composition.

5. Prime-Controlled Scaling Laws and Falsifiable Signatures

A structural reinterpretation lives or dies by what it predicts that other interpretations do not. Prime-only summation as an “irreducible aggregation” principle is not interesting if it merely restates known asymptotics of primes. It becomes interesting when it generates falsifiable signatures in systems where integer labels represent composed configurations and primes represent independent degrees of freedom. In that case, prime-controlled scaling laws are not optional; they are the measurable footprints of a generator–configuration architecture. This section collects those footprints into a practical diagnostic set: independent mode counts, log-log divergences, systematic differences between prime-indexed and integer-indexed aggregation, robustness to reweighting and windowing, and explicit criteria for distinguishing multiplicative composition from additive or random generative schemes.

Throughout, the point is not to demand that every domain reproduce textbook prime asymptotics exactly. The point is to identify scaling behaviors that are hard to fake unless a system truly has a sparse generator basis with multiplicative composition rules.

5.1 Independent mode counts: $N/\log N$ as a structural fingerprint

If primes are the independent modes and composites are configurations, then the number of degrees of freedom accessible up to a scale cutoff should grow like the number of primes up to that cutoff, not like the number of integers. This yields the first and simplest signature: mode-count growth is sublinear relative to the naive configuration count.

Let $M(x)$ denote the count of independent generator modes available up to scale x . In the literal integer world, $M(x)$ is the prime-counting function $\pi(x)$. Its leading behavior is approximately $x/\log x$. The key observation is that $x/\log x$ is neither linear nor constant; it is a distinctive “almost linear, but thinned” growth.

Operationally, this translates into a test:

If you can define a scale parameter x such that “generators up to x ” are meaningful, then count how many distinct irreducible modes appear up to that scale. If that count grows roughly like $x/\log x$ (or like a similarly thinned growth law), you have evidence that the system’s basis is sparse and prime-like.

This can be made concrete in many domains by defining a notion of irreducibility and a scale ordering. For example:

In a coded system, irreducible modes might be primitive symbols or minimal components, ordered by length or cost.

In a network failure system, irreducible modes might be independent root causes ordered by cost-to-activate or frequency.

In a compositional text or event stream, irreducible modes might be minimal factors under a chosen decomposition, ordered by rank or complexity.

The signature is not that the exact coefficient matches number theory. The signature is that the growth is systematically sublinear in a way that resembles thinning by a slowly varying factor, rather than being linear in x or exploding combinatorially.

Two refinements make the fingerprint sharper.

First, compare the generator count to the configuration count at the same scale. If configurations grow roughly like x while generators grow like $x/\log x$, then the ratio configuration/generator grows like $\log x$. This “log gap” is itself a signature: the number of composed states per generator grows slowly but unboundedly.

Second, look for scale invariance in the ratio after appropriate normalization. For primes, $\pi(x) \log x / x$ approaches a constant in the simplest approximation. In real systems, you look for stabilization of $M(x)$ times a slowly varying correction divided by x . Stabilization indicates a consistent thinning law rather than an arbitrary cutoff artifact.

5.2 Log-log divergence signatures and what they imply operationally

Some of the most diagnostic prime-controlled signatures are not mode counts but slow divergences. A classic example in prime mathematics is that the sum of reciprocals of primes diverges, but only as $\log \log x$. The key feature is not the specific function $1/p$; it is the existence of a divergence that is so slow that it is invisible at small scale but becomes unmistakable with enough dynamic range.

In a generator–configuration model, log-log divergences typically appear in observables that count “weak contributions” from many generators. For example, consider an observable that weights each generator by a decreasing function of its scale, such as $w(p)$ roughly proportional to $1/p$ or $1/e(p)$. The total contribution up to a cutoff can grow like $\log \log \text{cutoff}$. This indicates that infinitely many generators contribute, but each contributes very little; the aggregate grows slowly because the generator set is sparse.

Operational implications of a log-log signature are strong:

First, it suggests the system has no hard finite generator basis. New generators keep appearing as scale increases, but at a thinning rate.

Second, it implies that “tail management” matters. Even though each high-scale generator contributes little, the collective tail can still move the observable over large ranges of scale. Systems exhibiting such behavior are sensitive to long-range complexity even when dominated by low-scale modes at any fixed cutoff.

Third, it gives a practical falsification criterion. Many additive or random models produce either convergent sums (saturating) or polynomial divergences (fast growth). Log-log growth is rare and indicates a delicate balance: sparse mode density combined with slowly decaying weights.

One must be careful, however. A log-log trend can be mimicked by narrow windowing or by mixing different regimes. That is why the robustness tests in later subsections are essential: a true prime-like log-log signature should persist across changes in window size and weighting in predictable ways.

5.3 Comparing prime-aggregation scaling to integer-aggregation scaling

The generator–configuration thesis becomes falsifiable when it predicts systematic differences between prime-indexed aggregation and integer-indexed aggregation. This is not a contest over which is “better.” It is a test of whether prime-only aggregation is measuring a different object: independent degrees of freedom rather than the full configuration population.

A general method is to compute the same style of statistic under two aggregations:

Generator aggregation: $G(x) = \text{sum over primes } p \leq x \text{ of } g(p)$

Configuration aggregation: $C(x) = \text{sum over integers } n \leq x \text{ of } h(n)$

Then ask whether C can be reconstructed from G via a composition operator consistent with the model. In the Euler-product framework, configuration partition functions and their logarithms offer canonical examples: log configuration totals become generator sums.

At the scaling level, differences appear immediately:

If you sum a roughly constant function over integers up to x , you get growth like x . If you sum the same roughly constant function over primes up to x , you get growth like $x/\log x$. That gap is the generator–configuration distinction in its simplest expression.

But the deeper comparison is for domain-relevant observables. In an information system, for instance, a configuration-level “complexity” might grow rapidly because there are many composed objects, while a generator-level “novelty” grows slowly because new irreducibles are rare. If you mistakenly treat configuration growth as novelty growth, you will conclude that the system is generating independent new information faster than it really is.

Thus the comparison itself is a falsifiable claim:

If prime-only aggregation is the correct novelty measure, then configuration-level growth should decompose into a generator-level growth plus a composition growth. If configuration growth and generator growth track each other too closely, then the prime-generator ontology is likely wrong for that domain.

5.4 Robustness under reweighting and changes of observational window

Any scaling law can be “fit” over a narrow range. The framework becomes credible only if its signatures are robust under reasonable changes in how one observes and weights the system.

Two robustness principles are central.

First, reweighting robustness. If primes are truly independent generators, then many macroscopic observables should be expressible as sums over primes of functions of $e(p)$ or of weighted prime measures. Changing the weight function changes the observable in predictable ways. For instance, emphasizing small primes should increase occupancy-related observables without changing the qualitative thinning of generator density; emphasizing large primes should suppress most contributions and reveal whether large-scale generators exist in the data at all.

Second, windowing robustness. If one shifts the observation window from $[1, x]$ to $[x, 2x]$, prime density changes slowly. Thus generator counts and many weighted sums should change smoothly with scale, and ratios corrected by slowly varying factors should remain stable. If the observed signatures appear only in one special window but disappear under modest shifts, that suggests artifacts rather than structure.

This robustness requirement is also where the framework earns the right to claim “structural reinterpretation.” A structural law should be stable under observational choices that do not change the underlying ontology. If it is not stable, the ontology is likely not being measured.

Practically, one can propose a simple battery:

- Compute mode-count growth under multiple cutoffs.

- Compute weighted generator sums under multiple weights.

- Check whether normalized ratios stabilize.

- Check whether inferred spectra $e(p)$ are consistent across windows.

Consistency is evidence. Inconsistency is not a minor flaw; it is a direct falsification of generator independence or of the proposed encoding.

5.5 Distinguishing multiplicative composition from additive or random models

The final goal of this section is to show that the prime-controlled signatures are not generic. They are specific to multiplicative composition with irreducible generators. Competing generative schemes produce different fingerprints.

Consider three broad alternatives.

Additive composition models. Here complexity is built by addition of components rather than multiplication. The natural irreducibles are not primes; they are additive atoms, and the counting measure is typically linear in scale. Additive models often produce different tail behavior: counts grow like x , and weighted sums diverge like $\log x$ or as powers, not $\log \log$, unless fine-tuned.

Random labeling models. Here integers are labels without compositional meaning. Any apparent prime structure is an artifact. In such models, prime-only and integer-only aggregations are just different subsamples, and there is no reason for Euler-product reconstruction to hold. Inference of a stable $e(p)$ across windows should fail; robustness tests should break.

Mixed or coupled models. Here some generator-like structure exists but with interactions among generators. This is a realistic intermediate case. The key prediction is that Euler-product factorization breaks: prime occupancies become correlated, and $\log Z$ is no longer a simple sum of independent contributions. In data, this appears as systematic deviations from independent occupancy laws and as failures of reconstruction from prime-level spectra.

Therefore, the framework suggests a discriminating strategy:

If prime-only signatures hold and are robust, multiplicative generator structure is plausible.

If signatures hold only weakly and factorization breaks, the system may have generators but also interactions.

If signatures do not hold or are not robust, additive or random explanations are more plausible.

This is the point where “prime-only summation as ontology” becomes an empirical claim. It asserts that the world of interest has irreducible generators and multiplicative composition, and that the correct aggregation for independent structure lives on the generator set. The scaling laws and signatures described here provide a way to test that claim rather than merely admire it. The next section will formalize these ideas into an irreducible aggregation principle and show how configuration-level totals are reconstructed from generator-level data, clarifying exactly where the novelty of the approach lies and what it adds beyond standard analytic number theory.

6. The Irreducible Aggregation Principle: A General Counting Theorem

The previous sections built intuition: primes are generators, composites are configurations, and prime-only aggregation can be structurally correct when the object of interest is “independent degrees of freedom.” This section condenses that intuition into a general theorem-like statement: a criterion for when prime-only is the correct primitive sum, how configuration totals are reconstructed from generator data, and what kinds of claims the framework can and cannot support. The goal is not to replace analytic number theory, but to make a clean bridge from algebraic structure (irreducible generators) to measure-theoretic structure (how you should sum). The outcome is an irreducible aggregation principle: a statement about what deserves to be counted once, and what must be treated as a derived multiplicity.

6.1 Formal statement: when prime-only is the correct primitive sum

We begin with a general setting that includes the integers as a special case.

Assume we have a set of generators G and a set of configurations C formed by finite multisets of generators, with a composition operation that is commutative and associative. In the integers, G is the set of primes, C is the set of positive integers, and composition is multiplication.

Each configuration c in C has a unique generator-multiplicity representation:

c corresponds to an occupancy function $a_g(c)$ in $\mathbb{Z}_{\geq 0}$ for each g in G , with only finitely many nonzero.

Uniqueness is the key. It means configurations contain no new independent directions beyond generators. There is a well-defined mapping from configurations to occupancy vectors.

Now define a target functional Q that is intended to measure irreducible contribution. Examples include novelty, independent mode content, independent causes, or independent capacity. The irreducible aggregation principle states:

If Q is additive over independent generators and does not increase when generators are merely re-combined into configurations, then Q must be aggregated over generators, not configurations.

To make that statement formal, assume Q satisfies two properties.

First, generator additivity:

$$Q(c) = \sum_{g \in G} a_g(c) * q(g),$$

for some generator-level contribution $q(g)$.

Second, no overcounting by recombination:

There is no additional term $r(c)$ that would treat distinct compositions of the same generator multiset as adding new irreducible content. In other words, Q depends only on occupancy, not on configuration identity beyond occupancy.

Under these assumptions, prime-only (generator-only) summation is the correct primitive sum for Q . Any attempt to compute “total irreducible content up to a cutoff” by summing Q over configurations will systematically overcount because the same generator contributions are repeated across many configurations.

In the integer case, this becomes crisp:

If the irreducible content of n is intended to be the sum of contributions of its prime factors, then the primitive object is the prime contribution function $q(p)$. Summing over n is not wrong, but it is a different quantity: it is a sum over all multisets of primes. It counts configurations, not irreducible content.

Thus the formal criterion is structural and interpretive:

Prime-only is the correct primitive sum precisely when the quantity of interest is generator-additive and when composites are understood as emergent configurations rather than as new primitive contributors.

6.2 Reconstruction of configuration totals via composition operators

Once we accept that the generator-level contributions are primitive, the natural question is whether we lose information by refusing to sum over configurations. The answer is no, because configuration totals are reconstructible from generator data via composition operators. This is the second half of the principle: “count generators once, then rebuild configurations.”

The simplest reconstruction operator is the partition function introduced earlier. Fix a generator cost $e(g)$ and define configuration cost by occupancy additivity:

$$E(c) = \sum_g a_g(c) e(g).$$

Then the configuration partition function is

$$Z(\beta) = \sum_{c \in C} \exp(-\beta E(c)).$$

Because occupancies are independent and unbounded, Z factorizes:

$$Z(\beta) = \prod_{g \in G} 1/(1 - \exp(-\beta e(g))).$$

This is the reconstruction statement: the configuration total Z is a deterministic functional of the generator spectrum $e(g)$. You do not need to treat configurations as primitive inputs. You derive them.

More generally, many configuration-level totals can be reconstructed from generator-level data by exponential and product transforms. The common pattern is:

Configurations correspond to multisets.

Multisets are generated by products of geometric series factors.

Logarithms turn products into sums over generators.

This shows why prime-only aggregation is not “throwing away” composites. It is relocating them. Composites reappear as the output of a composition operator rather than as objects of primitive summation.

In practical modeling terms, this gives a workflow:

Estimate or specify generator-level contributions and spectra.

Compute predicted configuration-level distributions or totals via a composition operator.

Compare the predictions to observed configuration statistics.

If reconstruction succeeds, generator primacy is supported. If it fails, the model is structurally wrong or incomplete.

6.3 Relations to Euler products, logarithms of generating functionals

The irreducible aggregation principle is inseparable from a transformation that isolates generator content. For the integers, that transformation is the Euler product and its logarithm. But the deeper relation is universal: whenever a configuration space is generated as multisets of irreducibles, the generating functional factorizes.

At the level of products, the Euler-product form appears because each generator contributes an independent geometric series for its possible multiplicities. At the level of sums, the logarithm is the canonical operator that converts multiplicative composition into additive decomposition.

This clarifies a key conceptual point: in a generator–configuration system, the log of a configuration generating functional is often the most direct measure of generator content. It is the “additive shadow” of multiplicative composition.

In the integer case, one sees this in the relationship between:

The configuration sum over n of $n^{-\beta}$.

The product over primes of $1/(1 - p^{-\beta})$.

The log of that product, which becomes a sum over primes of $\log(1 - p^{-\beta})^{-1}$.

That last object is the generator decomposition. It is exactly the kind of sum a generator ontology would privilege.

Therefore, the principle can be restated in a way that emphasizes transform structure:

If your object is generated by multiplicative composition of irreducibles, then the correct primitive aggregation is the one that linearizes composition. For integer multiplication, logarithms do this, and prime-only sums are the irreducible index set that survives this linearization.

This is also where the framework gains portability: “prime-only” is really a special case of “irreducible-only” under whatever composition law defines your domain. If your domain has its own notion of irreducibles under a composition operation, the same logic applies.

6.4 Error terms and what can and cannot be controlled

A counting principle becomes dangerous if it encourages overconfident claims. Here we delineate what is controllable and what is not.

What is controllable, in principle, is reconstruction error within the model class. If you assume independent generator occupancies and additive cost, then configuration totals derived from the Euler product are exact within that model. Any mismatch with data is then attributable to violations of assumptions: generator interactions, constraints, non-multiplicative composition, or an incorrect encoding of domain objects as multiplicative configurations.

What is generally not controllable without additional structure are fine-grained configuration-level errors when only coarse generator statistics are known. Knowing the generator spectrum $e(p)$ allows you to compute ensemble averages, but not necessarily the exact frequency of each configuration in a finite dataset unless the data truly is sampled from the implied ensemble.

Another limitation concerns identifiability. Multiple distinct spectra can produce similar configuration totals over restricted ranges. This means one must be cautious about claiming uniqueness of $e(p)$ based on limited observations. Robustness across multiple conditions, cutoffs, or weightings is the antidote.

A further limitation is that not every meaningful observable is generator-additive. Some phenomena are inherently interaction-based: the “meaning” of a composite might be more than the sum of its parts. In such cases, prime-only aggregation is not wrong, but it is incomplete: it measures irreducible availability, not emergent interaction content. The principle therefore does not deny emergence. It separates irreducible accounting from emergent accounting and insists that they not be conflated.

Finally, there is a subtle but important interpretive constraint: the encoding must be faithful. If integers label observations arbitrarily, prime structure is meaningless. The principle applies only when the decomposition into irreducibles corresponds to something real in the domain, not just in the numbering scheme.

In short, the principle provides exact results inside its assumptions and provides falsification signals when those assumptions fail. It does not promise magical extraction of truth from any dataset with integer labels.

6.5 Conceptual payoff: “primes as basis” without metaphor

The phrase “primes are a basis” is often used loosely. Here it can be stated without metaphor.

A basis, in a linear setting, is a set of independent generators such that every element of the space is uniquely expressible as a combination of basis elements. The integers under multiplication form a commutative monoid, not a vector space, but the same structural idea applies: primes are independent generators, and every integer has a unique expression as a finite multiset of primes with multiplicities. The occupancy vector plays the role of coordinates.

The irreducible aggregation principle says: if you want to measure independent structure, measure at the coordinate-axis level, not at the point level. Points in coordinate space are configurations; axes are generators. Summing over points answers a different question than summing over axes. That is the basis claim in operational form.

The payoff is conceptual hygiene. It prevents a common mistake: interpreting configuration proliferation as generator proliferation. Composites proliferate quickly because there are many ways to combine generators. That combinatorial explosion does not imply new irreducibles are being created. Prime-only aggregation, treated as primitive, is the mathematical discipline that keeps these two kinds of growth distinct.

This is the heart of the “ontology shift.” Prime sums are no longer just a convenient subset of integer sums. They are the correct measure of irreducible degrees of freedom in multiplicatively compositional systems. And the full configuration world is not discarded; it is reconstructed from irreducibles by explicit composition operators. That is the general counting theorem in its practical form: count irreducibles once, and treat composites as emergent multiplicities generated by composition.

7. An Information-Theoretic Novelty Functional Built on Primes

If prime-only summation is to earn the status of “correct aggregation” outside pure number theory, it must produce useful observables: quantities that can be computed from real datasets and that separate independent novelty from combinatorial recombination. This section introduces one such observable family: prime-novelty and prime-surprise functionals. The core idea is simple: treat primes as irreducible features (generators) and treat composites as composed descriptions whose informational content largely derives from already-seen generators. A novelty functional should therefore count generator activation, not configuration

proliferation. Composites may still matter, but they should contribute as interaction terms, not as repeated claims of “new independent content.”

The prime-novelty framework does not assume that real-world data “is” number theory. Instead it assumes a representation step: observations are encoded into integers so that multiplicative composition corresponds to composition of underlying factors. When such an encoding is justified, prime factorization becomes a principled feature extractor, and prime-only aggregation becomes a principled novelty measure. The payoff is operational: you can compute novelty curves, test scaling laws, and compare multiplicative generative models to additive or random alternatives.

7.1 Defining prime-novelty and prime-surprise for observed datasets

Let a dataset D consist of observations $x_1, x_2, \dots, x_T$. Assume each observation is mapped to a positive integer n_t by an encoding ϕ such that the factor structure of n_t is meaningful for the domain. Write the prime factorization of each n_t as:

$$n_t = \prod_p p^{a_p(t)},$$

where $a_p(t)$ is the exponent of prime p in the t -th observation’s encoding.

From these exponents we define two families of prime-based quantities.

First, the prime activation indicator for a prime p at time t :

$$I_p(t) = 1 \text{ if } a_p(t) > 0, \text{ and } 0 \text{ otherwise.}$$

This is the simplest notion of “did generator p appear at all in this observation?”

Second, the prime multiplicity $a_p(t)$ itself, capturing repeated use of the same generator within an observation.

A novelty functional should convert the history of activations and multiplicities into a scalar score describing how much new irreducible structure is being introduced as the dataset evolves.

A clean starting point is the cumulative distinct-prime count:

$$N_{\text{distinct}}(T) = \text{number of primes } p \text{ such that there exists } t \leq T \text{ with } I_p(t)=1.$$

This measures how many unique irreducible generators have ever appeared. It is the analog of “vocabulary size” in language, but with prime factors as vocabulary items.

However, N_{distinct} is too coarse for many purposes because it counts a prime once and then ignores frequency. We therefore introduce prime-novelty as a weighted sum over primes:

$$\text{PrimeNovelty}(T) = \sum \text{over primes } p \text{ of } w(p) * g(C_p(T)),$$

where $C_p(T)$ is a count statistic such as:

$$C_p(T) = \sum_{t=1..T} I_p(t) \text{ (how many observations contained prime } p),$$

or alternatively:

$$C_p(T) = \sum_{t=1..T} a_p(t) \text{ (total multiplicity across observations),}$$

and where g is a concave function that increases with C but with diminishing returns (so repeated sightings of the same prime add little novelty), and $w(p)$ weights primes by scale or importance.

A closely related quantity is prime-surprise: how unexpected is it to see prime p at time t given its past frequency? Define an empirical probability estimate for prime presence:

$$P_{\hat{p}}(t-1) = (C_p(t-1) + \alpha) / (t-1 + 2\alpha),$$

with a small smoothing $\alpha > 0$. Then define the surprise contribution of p at time t as:

$$S_p(t) = -\log P_{\hat{p}}(t-1) \text{ if } I_p(t)=1, \text{ else } 0.$$

$$\text{PrimeSurprise}(t) = \sum_p w(p) * S_p(t).$$

This behaves like a weighted self-information score: rare primes contribute more surprise; common primes contribute little. Over time, as a prime becomes common, its surprise contribution decays automatically. This provides an adaptive novelty measure that aligns with the intended semantics: novelty is not “new numbers,” it is “new generators or rare generator activations.”

7.2 Divisibility statistics and expected prime signatures under mixing

To use prime-novelty diagnostically, we need baseline expectations: what prime activation rates should look like under simple generative assumptions?

A convenient baseline is a mixing model: observations are formed by combining a random subset of available generators, possibly with multiplicities, and then encoding the combination multiplicatively. Under such a model, divisibility statistics become the central observables: how often is a random observation divisible by prime p , or by p^2 , and so on?

In the simplest case, assume each prime p is independently included in an observation with some probability r_p , and if included, has multiplicity drawn from some distribution. Then:

$$\Pr[I_p = 1] = r_p,$$

and

$$E[I_p] = r_p.$$

If multiplicities are geometric with parameter depending on $e(p)$ and β (as in the partition ensemble), then:

$$E[a_p] = 1 / (\exp(\beta e(p)) - 1),$$

and

$$\Pr[l_p = 1] = 1 - \Pr[a_p = 0] = 1 - (1 - \exp(-\beta e(p))) = \exp(-\beta e(p)).$$

This relationship is important: it ties presence probability directly to generator energy. Low-energy primes are almost always present; high-energy primes are rarely present. Thus the activation spectrum across primes is a map of $e(p)$ filtered through β .

Even without committing to a full partition ensemble, the key is that a multiplicative mixing process produces characteristic divisibility signatures:

Small primes appear frequently.

Large primes appear rarely.

The decay of appearance with prime size encodes the generator spectrum.

Higher powers p^k appear with rapidly decreasing frequency, reflecting multiplicity rules.

A diagnostic workflow follows:

Compute empirical presence rates $\hat{r}_p$ = fraction of observations divisible by p .

Compute empirical power rates $\hat{r}_{p^k}$ = fraction divisible by p^k .

Fit a candidate spectrum $e(p)$ and occupancy rule to match these rates.

Test whether the fitted spectrum is stable across observation windows.

If stability holds, the model has teeth. If it does not, the prime encoding may be wrong, generator independence may fail, or the dataset may not be produced by multiplicative mixing at all.

7.3 Overcounting corrections: composites as redundant descriptions

Now we come to the conceptual core: why do we want a prime-novelty functional at all?

Because naive novelty measures over configurations overcount. If you treat each distinct integer n as “new information,” novelty grows quickly simply because there are many composites. But that is combinatorial proliferation, not irreducible innovation.

Prime-novelty corrects this by treating composites as redundant descriptions unless they introduce new prime factors (new generators) or activate rare primes (surprise).

To see the correction, consider two sequences of observations.

Sequence A repeatedly combines the same small set of primes in many new ways: 23, 235, 237, 2^{23} , 2^{335} , and so on. The set of distinct integers grows rapidly. But the set of distinct primes might grow slowly or not at all after a point.

Sequence B introduces new primes steadily: 2, 3, 5, 7, 11, 13, and so on.

A configuration-based novelty measure would say both sequences are generating lots of novelty because both produce many different integers. A prime-based novelty measure distinguishes them: Sequence A is largely recombination; Sequence B is irreducible expansion.

In informational terms, composites are redundant because their prime factors can be inferred from earlier observations once the generator set is known. They are new configurations, but not necessarily new irreducible content. This does not deny emergent meaning; it separates emergent structure from generator novelty. If one wants to measure emergence, one can introduce explicit interaction terms, such as novelty of co-occurrence patterns beyond what generator frequencies predict. But that is a different layer.

The overcounting correction therefore has a clear semantics:

Generator novelty counts newly activated basis elements.

Configuration novelty counts newly encountered combinations.

Both may matter, but they should not be conflated.

Prime-novelty is the generator novelty layer.

7.4 Practical estimators and finite-sample behavior

In real datasets, prime-novelty must be estimated with finite samples and limited dynamic range. This introduces practical considerations.

First, rare primes may not appear at all in a finite dataset, even if they are part of the true generator set. Thus $N_distinct(T)$ underestimates the true accessible generator count. This is not a bug; it is a finite-sample reality. One can address it using capture–recapture style estimators or by fitting an activation spectrum and extrapolating.

Second, surprise-based measures require probability estimation. Empirical frequency estimates for rare events are noisy, so smoothing is essential. The smoothing parameter α should be chosen so that early-time surprises do not explode artificially.

Third, weighting choices matter. If $w(p)=1$, all primes contribute equally. If $w(p)=\log p$, larger primes contribute more per activation, reflecting that they carry larger scale or complexity. If $w(p)$ is chosen as a decreasing function such as $1/p$, then small primes dominate and the functional becomes sensitive to the “core generator set” rather than to rare large primes.

Different choices correspond to different questions:

Do you want to count distinct irreducibles regardless of size? Use $w(p)=1$.

Do you want novelty weighted by multiplicative scale? Use $w(p)=\log p$.

Do you want novelty emphasizing foundational generators? Use decreasing weights.

Fourth, finite-sample windowing matters. The novelty curve often has two regimes: an initial rapid rise as common primes are discovered, followed by a slower tail as rare primes appear. That tail behavior is where prime-like scaling signatures may emerge, but it requires enough data to observe.

Finally, computational issues are manageable in many settings because factorization can be done efficiently for moderate-size integers or by storing prime signatures directly if the encoding is constructed from known factors. If the integers are large, one can use partial factorization, probabilistic factorization, or domain-specific factor extraction in place of full prime factorization, as long as the extracted factors behave as irreducibles under the domain's composition rule.

7.5 Predictions: how novelty grows under multiplicative generative dynamics

The point of defining a novelty functional is not merely to compute it, but to predict its growth under different generative dynamics.

Under multiplicative generative dynamics with a stable generator spectrum, prime-novelty growth typically shows:

A fast initial regime where low-cost primes rapidly become common and surprise decays quickly for them.

A long tail regime where new primes appear slowly and surprise contributions remain intermittent.

A scaling signature in the distinct-prime count that is sublinear and often compatible with thinning laws like $x/\log x$ when plotted against an appropriate scale parameter.

More specifically, if observations are drawn from a prime-factorized ensemble governed by $e(p)$ and β , then:

Presence rates should follow $\exp(-\beta e(p))$ approximately.

Multiplicity rates should follow $1/(\exp(\beta e(p)) - 1)$ approximately.

The distribution of discovered primes over time should be stable under window shifts, once corrected for finite-sample effects.

Surprise spikes should correspond to activation of higher-cost primes, and the statistics of spikes should be predictable from the tail of $\exp(-\beta e(p))$.

Crucially, these predictions differentiate multiplicative mixing from additive or random labeling.

Additive generation tends to produce different novelty profiles: either more linear vocabulary growth or different tail behavior, depending on the additive atom structure.

Random labeling produces instability: any inferred prime spectrum depends on the arbitrary mapping and will not remain stable under changes of observation window or reweighting.

Therefore, prime-novelty is not just a measurement; it is a model diagnostic. It can be used to infer whether a dataset’s apparent complexity is primarily due to recombination of a fixed generator basis or due to ongoing introduction of new irreducibles.

This concludes the main construction. The next section will build on it to propose explicit model selection diagnostics and falsification protocols: how to compare prime-based novelty models to alternative explanations, how to detect generator interactions, and how to decide whether “prime-only” is truly the correct primitive aggregation for a given domain rather than a projection imposed by the analyst.

8. Model Selection and Diagnostics: Testing for Multiplicative Generative Structure

A structural reinterpretation is only as strong as its ability to be wrong. The prime-generator ontology proposed in this paper makes concrete claims: that the system’s independent degrees of freedom are irreducible, that composition is fundamentally multiplicative (or representable as such), that generator contributions are additive on prime exponents, and that configuration totals are reconstructible by Euler-product-like operators. These claims generate signatures, but they also generate failure modes. This section turns the framework into an empirical protocol: define competing hypotheses, compute diagnostic curves, stress-test the diagnostics against noise and bias, establish null baselines, and state explicit falsification criteria. The goal is to make “prime-only is the correct primitive sum” an operational model selection statement rather than a philosophical preference.

8.1 Competing hypotheses: additive generation, random labeling, mixed regimes

The prime-generator view is not the default in most data settings. Before adopting it, one should specify alternatives that are plausible for the domain.

A first alternative is additive generation. Here observations are formed by addition of independent components rather than multiplication. Under such a model, the natural irreducibles are additive atoms, and the canonical decomposition is into sums, not products. If you encode such a system multiplicatively, prime factorization becomes an awkward coordinate system: it may still produce patterns, but those patterns need not be stable or meaningful.

A second alternative is random labeling. Here integers are merely identifiers. Any prime structure is an artifact of the label assignment, not an intrinsic feature of the system. Under random labeling, prime-based diagnostics should fail robustness tests: inferred spectra should not be stable across windows, and reconstruction from prime statistics should not generalize.

A third alternative is a mixed regime: there is some multiplicative composition, but not purely. Perhaps some generators combine additively, others multiplicatively; perhaps there are constraints, couplings, or exclusion rules that introduce correlations among generators. This is common in real systems. In this regime, prime-based methods may still be useful, but the clean Euler-product factorization will break, and one should expect systematic interaction terms.

A fourth alternative is multiplicative generation with interactions. This is a special case of mixed regimes where composition is multiplicative, but generator independence fails. In physical terms, the prime modes are “interacting,” so the partition function is not a simple product. In informational terms, some irreducible factors co-occur more or less than independence would predict.

The framework’s diagnostic stance is therefore:

Prime-generator ontology is plausible if multiplicative structure is present and generator independence is at least approximately valid, or if deviations from independence can be modeled as interaction corrections. It is implausible if additive or random-label models explain the data equally well without needing prime structure, or if prime-based results are not robust.

8.2 Diagnostic plots: scaling curves and response functions

The most reliable way to distinguish hypotheses is to avoid single statistics and instead examine families of curves whose shapes are hard to fake. The prime-generator framework offers several such curve families.

First, distinct-prime growth curves. Plot the cumulative number of distinct primes observed as a function of time or scale. Under a stable multiplicative generator process, the curve rises quickly at first and then slows, reflecting that common primes are discovered early and rare primes arrive late. Under random labeling, the curve’s shape depends strongly on the arbitrary mapping and often fails window stability. Under additive generation, the curve may behave differently, often closer to linear growth in the number of atoms discovered, depending on the system.

Second, prime activation spectra. For each prime p in a chosen range, estimate its presence probability r_{hat}_p and plot r_{hat}_p against p or against a scale proxy such as $\log p$. Under generator-energy models, r_{hat}_p should follow a smooth monotone envelope consistent with

$\exp(-\beta e(p))$ for some $e(p)$. Deviations from smoothness can indicate interactions, multiple regimes, or encoding artifacts.

Third, prime multiplicity spectra. Estimate the rate of p^2 , p^3 divisibility. Under independent occupancy models, these higher-power rates have a predictable relationship to $\hat{r}_p$. Large systematic deviations indicate that the multiplicity model is wrong or that composition does not behave like independent occupancy.

Fourth, response functions under reweighting. Define weighted prime sums such as $\sum_p w(p) \hat{r}_p$ for various w . Under a prime-generator ontology, changing w should shift emphasis in predictable ways and should not produce inconsistent inferences about the underlying spectrum. If slight changes in w wildly change conclusions, the inferred structure is likely unstable.

Fifth, reconstruction checks. Use inferred generator spectra to predict configuration-level totals or distributions under the model and compare with observed configuration statistics. For instance, if you fit $e(p)$ and β from prime activation rates, you can predict expected distributions of configuration energies $E(n)$ or expected complexity measures derived from occupancy. If predictions systematically miss, the model is wrong, regardless of how pretty the prime plots look.

In all cases, the goal is not to force the data to look like primes. The goal is to detect whether the data behaves as though it is generated by a sparse set of irreducibles with a stable activation spectrum and multiplicative recombination.

8.3 Sensitivity to noise, missingness, and observational bias

Real datasets are messy. Prime-based diagnostics can be sensitive in distinctive ways, so it is essential to test robustness under realistic perturbations.

Noise in the encoding. If the mapping from observations to integers is approximate, errors may introduce spurious prime factors or remove real ones. This can inflate distinct-prime counts and create artificial tail behavior. A practical guardrail is to measure how often “rare primes” appear as singletons. A large number of primes that appear exactly once can indicate encoding noise rather than genuine generator discovery. Another guardrail is to compare factorization stability under repeated encodings or under known perturbations of the observation.

Missingness. If observations are missing in a non-random way, prime activation rates can be biased. For example, if high-complexity observations are under-sampled, large primes may appear artificially rare. A standard correction is to stratify by observation size or by a proxy for complexity and compute activation rates within strata.

Window bias. If you only observe a short time window in a nonstationary system, you may infer a spectrum that reflects a local regime rather than a stable generator law. The remedy is to compute diagnostics across multiple windows and check whether inferred parameters are consistent.

Censoring and cutoffs. If the system has a hard cutoff that excludes large generators (e.g., maximum code length, maximum complexity, finite resolution), then prime signatures in the tail will saturate. This is not a failure of the model; it is a constraint to be modeled explicitly. But if you misinterpret saturation as convergence or as absence of new generators, you will draw the wrong conclusion about the generator basis.

Interaction with sampling. Because prime density thins with scale, sparse sampling can mimic prime thinning. A dataset with fewer samples at higher scale can produce an apparent $1/\log$ effect even without true prime-like structure. This is why diagnostics must compare within-scale rates, not just overall counts, and must use null models that preserve sampling intensity across scale.

8.4 Null models and permutation baselines

A convincing diagnostic requires baselines: models that preserve superficial features of the data while destroying the hypothesized structure. Prime-generator claims should survive against these baselines; if they do not, they were likely artifacts.

A simple baseline is label permutation. Keep the sequence of observations but randomly permute the integer labels assigned to them. This preserves the distribution of labels but breaks any meaningful relation between observation content and factor structure. If prime-based novelty and spectra remain “similar” after permutation, then the observed prime structure was not content-driven.

A second baseline is factor scrambling. Replace each observation’s prime factors with randomly chosen primes while preserving some marginal constraints, such as the total number of factors per observation or the total energy $E(n)$. This preserves configuration complexity but destroys specific generator identities. If your model’s inference depends on specific primes being special rather than on the overall spectrum shape, this baseline will reveal it.

A third baseline is additive-encoding null. Encode observations using an additive scheme (or a deliberately non-multiplicative mapping) and compute the same prime-based diagnostics. If “prime structure” appears similarly strong under non-multiplicative encodings, that indicates that the diagnostics are responding to generic heavy tails or sampling artifacts rather than to multiplicative composition.

A fourth baseline is block-shuffling in time. If you suspect nonstationarity, shuffle observations within blocks to destroy long-range trends while preserving local distributions. True generator spectra should be relatively stable under such shuffling if the process is stationary; if not, the apparent spectrum may be driven by temporal drift.

The key is that null models should preserve what you are not trying to test (marginal distributions, counts, cutoffs) and destroy what you are trying to test (meaningful multiplicative generator structure). If prime-based signatures survive, they are more credible.

8.5 What would falsify the prime-generator ontology

It is important to state explicit falsification criteria. Otherwise, any dataset can be “interpreted” as prime-like with enough creativity.

The prime-generator ontology is falsified for a given domain if one or more of the following occur robustly:

First, encoding instability. There is no encoding ϕ that is both domain-justified and yields stable prime signatures. Small changes in encoding produce large changes in inferred generator spectra or novelty curves. This indicates that primes are artifacts of representation rather than irreducibles of the system.

Second, window instability. Inferred prime activation spectra and fitted $e(p)$ vary substantially across observation windows in ways that cannot be explained by known nonstationarity or by changes in control parameters. A genuine generator spectrum should be stable or should drift in an explainable manner.

Third, failure of reconstruction. You can fit prime-level activation rates well, but configuration-level predictions derived from the model systematically fail. This indicates that the model is overfitting prime marginals while missing the true composition law or interactions.

Fourth, lack of factorization. Measured co-occurrence statistics show strong, persistent correlations among primes that cannot be captured by small interaction corrections. In such cases, a pure independent-generator Euler-product model is wrong. One might still have irreducibles, but not independent ones, and the “prime-only” aggregation may not be the correct primitive measure for the intended observable.

Fifth, null-model indistinguishability. The prime-based signatures are not significantly different from those obtained under label permutation, factor scrambling, or other null baselines that destroy meaningful factor structure. If the signatures persist under nulls, they are not evidence of a prime-generator ontology.

Sixth, additive superiority. An additive or non-multiplicative model explains the data with equal or better predictive performance and yields more stable, interpretable irreducibles. In such a case, insisting on primes is unnecessary and likely misleading.

These falsification criteria keep the framework honest. They also clarify the intended scope: the prime-generator ontology is not a universal claim about all data. It is a claim about a class of systems that are genuinely compositional under multiplication or under an isomorphic composition rule. In that class, prime-only aggregation is not an arbitrary choice but an appropriate measure of irreducible structure. Outside that class, the framework should fail cleanly, and the diagnostics above are designed to make that failure unmistakable.

9. Applications and Interpretive Domains

The prime-generator framework is deliberately abstract. Its core claim is structural: when a system has irreducible generators and multiplicative (or multiplicatively representable) composition, prime-only aggregation is the correct primitive measure for independent structure, while configuration-level behavior should be reconstructed through composition operators. This section grounds that claim in interpretive domains where (a) generator–configuration architecture is natural, (b) overcounting through recombination is a known problem, and (c) one can plausibly design tests that distinguish irreducible novelty from configuration proliferation. The intent is not to force every domain into prime clothing, but to identify where the framework naturally attaches and where it should be expected to fail.

9.1 Physical analogs: spectra, modes, and compositional state spaces

Physics supplies the cleanest intuition for generator spectra and partition functions, even when one has no intention of claiming that primes are literally “particles.” The point of contact is the idea of independent modes. Many physical systems are most naturally described not by enumerating all configurations, but by specifying the mode spectrum, then building configurations as occupations of modes.

In such a description, the generator set is the set of modes. A configuration is an occupation pattern across modes. Energies add across occupied modes. Partition functions factorize when modes are independent. This is precisely the structure formalized earlier, with primes playing the role of a canonical generator index set.

The prime-only concept becomes physically meaningful whenever the system exhibits a sparse emergence of new independent modes with increasing scale. For example, imagine a complex device whose effective degrees of freedom increase with scale, but not linearly: new independent vibrational modes, failure modes, or resonances become available only gradually

as the system grows. If the density of new modes thins with scale in a prime-like way, then a prime-indexed generator measure becomes a compact model: it predicts how many new independent degrees appear and how configuration diversity grows from them.

The most concrete physical use-case is not to replace Fourier modes with primes, but to use prime-like thinning as a signature of irreducible mode emergence. In systems where independence is the scarce resource and configurations are plentiful, prime-only mode counting and prime-controlled response curves provide a disciplined way to separate “more combinations” from “more true degrees of freedom.”

The compositional state space viewpoint is equally important. Many physical state spaces are huge because configurations are combinatorial products of modes. If one mistakenly counts configurations as independent degrees, one overestimates physical capacity and misinterprets entropy-like quantities. The prime-generator view guards against that mistake: it insists on tracking the mode spectrum first, then deriving configuration multiplicity as a consequence.

9.2 Informational systems: codes, logs, event streams, and description length

Informational domains are where the framework often becomes most directly actionable, because “irreducible novelty versus recombination” is a daily problem in real data.

Consider a system that emits events. The event stream can look endlessly novel because the exact event sequences are rarely repeated. But the underlying causes may be few: a small set of independent mechanisms recombine into many observable patterns. In such a setting, configuration-level novelty (new sequences) grows rapidly, while irreducible novelty (new independent causes) grows slowly. A prime-based novelty functional is designed to measure the latter.

Codes are an especially good match. Many coding systems are compositional: a message is assembled from primitives; higher-level tokens are products of primitives in a structural sense. If an encoding can map “composition” to multiplication and “primitive components” to primes, then factorization becomes a principled way to decompose messages into irreducible components. Novelty is then the introduction of new prime factors or the activation of rare ones.

Logs and telemetry offer another match. Many log lines are unique, but they are constructed from a limited vocabulary of templates, IDs, and parameters. A prime-based factor signature can represent template components as irreducibles and treat a full log event as a composite. The overcounting correction becomes meaningful: the system is not inventing new irreducible information every time a timestamp changes. It is recombining primitives.

Description length is the most general informational lens. In information theory, the “cost” of describing something is central. The generator spectrum $e(p)$ can be interpreted as the marginal description cost of introducing irreducible component p . The induced configuration cost $E(n)$ becomes the description length of a composed object. When description length is additive over independent components, the Euler-product thermodynamics becomes a rigorous way to generate distributions over composite descriptions.

The practical question in informational settings is always the same: does the prime-factor representation actually compress or explain the data better than alternatives? If it does, and if the inferred spectrum is stable across windows and robust to reweighting, then prime-only aggregation is not a flourish; it is measuring something real about the system’s irreducible structure.

9.3 Networks and systems engineering: independent failure modes and coupling

Engineering systems are full of combinatorial explosions. There are many ways a system can fail, many ways components can interact, and many observable symptoms. But independent root causes are often far fewer. This is exactly the distinction prime-only aggregation is meant to respect.

In a reliability model, one may care about:

How many independent failure modes exist up to a cost threshold?

How often do new independent modes appear as the system grows?

How do coupled modes produce cascades?

If independent modes are the primitive object, then a generator measure is appropriate. A prime-like thinning law becomes plausible in large engineered systems: the easiest modes are discovered early, and new independent modes become rarer as complexity increases, because most new observed failures are recombinations or cascades of known issues.

Here the mixed-regime discussion becomes important. Real engineering systems rarely have independent modes. They have coupling. The Euler-product factorization is therefore not expected to hold perfectly. But that is not a weakness; it is a diagnostic. Deviations from factorization quantify coupling strength. In practice, one might:

Fit an independent-generator model as a baseline.

Measure co-occurrence correlations among inferred irreducible factors.

Add interaction corrections where needed.

Prime-based novelty then becomes a way to track whether the system is experiencing genuinely new failure primitives or just new combinations of old ones. That distinction can guide

engineering prioritization: a rise in configuration novelty with stable generator novelty suggests recombination; a rise in generator novelty suggests new independent risk.

This domain also supplies a good reason to weight primes: $w(p)$ can represent severity, cost, or impact. A rare but catastrophic irreducible failure mode should contribute more novelty in an operational sense than a common minor mode.

9.4 Cognitive and semantic systems: irreducible causes versus derived narratives

Cognitive and semantic domains are slippery, but they are also where the distinction between irreducible causes and derived narratives is most consequential.

Human cognition and discourse generate an enormous number of distinct narratives, explanations, and interpretations. But the underlying causal primitives may be relatively few: repeated motifs, stable biases, a limited set of needs, fears, incentives, and constraints. The “combinatorial narrative space” can expand without an accompanying expansion of irreducible cause space.

The prime-generator framework offers a disciplined metaphor that can become operational if one defines an encoding of semantic elements into irreducible factors. For example, in a constrained ontology of causes, one can represent an explanation as a composite of irreducible cause tokens. The factorization then separates:

Which irreducibles are present at all (generator activation).

How often they appear (multiplicity).

How they combine (interaction and narrative structure).

Prime-only aggregation corresponds to measuring how many irreducible causes are being invoked. Configuration counts correspond to the number of distinct stories.

This distinction matters in interpretive work because it can prevent a common error: mistaking an explosion of discourse for an explosion of causes. The prime-novelty functional, when adapted to a semantic factorization scheme, becomes a tool for measuring whether a community is actually introducing new explanatory primitives or simply remixing existing ones into new rhetorical forms.

At the same time, this is also where the framework can fail if one overreaches. Semantic meaning is not always additive over parts. Interaction terms may dominate. Therefore, in cognitive and semantic applications, prime-only aggregation should be treated as a baseline measure of irreducible invocation, not as a complete account of meaning. The value is in separation: irreducible cause accounting first, emergent narrative structure second.

9.5 Guidance for choosing $e(p)$ and $w(p)$ in applied settings

Applied use hinges on two design choices: the generator energy $e(p)$ and the novelty weight $w(p)$. These are not arbitrary knobs; they encode the ontology of importance in the domain.

Choosing $e(p)$: $e(p)$ should reflect the marginal cost of activating generator p . In physics-like analogs, $e(p)$ is an energy level. In information-like settings, it is description length or complexity. In engineering settings, it may be cost-to-trigger or difficulty-to-discover. The key requirement is that configuration cost be additive over generator multiplicities, at least approximately. If additivity fails badly, the model must include interactions.

Practical approaches to $e(p)$ include:

Structural choice: $e(p)$ proportional to $\log p$ if the integer encoding is literal and multiplicative scale should map to cost.

Parametric choice: $e(p) = a \log p + b (\log p)^\alpha$ or $e(p) = p^\gamma$ when one expects stronger penalties for large generators.

Empirical choice: fit $e(p)$ from observed presence rates using the occupancy law, then test stability across windows.

Choosing $w(p)$: $w(p)$ determines what you mean by novelty.

If you want pure irreducible count, use $w(p)=1$ and a novelty function g that saturates quickly.

If you want novelty weighted by scale, use $w(p)=\log p$ or another increasing function.

If you want novelty weighted by operational impact, use domain-specific weights (severity, risk, cost, importance), potentially independent of p 's numeric size.

If you want foundational generator emphasis, use decreasing weights such as $1/p$ or $1/e(p)$ to focus on the core basis.

A useful discipline is to separate the two roles:

$e(p)$ shapes the generative model and the configuration ensemble.

$w(p)$ shapes the measurement objective and how novelty is valued.

Conflating them can cause interpretive confusion. For example, using $w(p)=e(p)$ may be appropriate when novelty is “energy introduced,” but inappropriate when novelty is “new irreducible types” regardless of energy.

Finally, any applied setting should follow the same principle: justify the encoding, estimate or choose $e(p)$, define $w(p)$ to match the measurement goal, compute prime-based diagnostics, and then test against null models and robustness checks. If the inferred generator structure is stable and predictive, prime-only aggregation has earned its role as the correct primitive sum in

that domain. If not, the framework should be discarded without sentimentality, because it has done its job: it has offered a falsifiable hypothesis about irreducible structure.

10. Limitations, Open Problems, and Research Program

The prime-generator ontology has been presented as a structural reinterpretation rather than a poetic metaphor: primes index irreducible degrees of freedom, composites are emergent configurations, and prime-only aggregation is the correct primitive sum when the target quantity is irreducible. That framing is powerful precisely because it is fragile. It depends on uniqueness of decomposition, on a faithful encoding from domain objects to multiplicative compositions, and on a separation between irreducible accounting and emergent interaction content. This section makes those dependencies explicit, identifies where the approach breaks, and then lays out a research program aimed at turning the framework into a practical methodology for identifying multiplicative generative structure in real systems.

10.1 Where the analogy breaks and where it becomes mathematics

There are two very different claims one can make with “prime-only summation.”

The weak claim is analogical: primes remind us of irreducibles, and prime sums remind us to count independent contributions rather than all combinations. This is often true but scientifically thin. Almost any sparse subset can play the role of “special indices” if you do not demand reconstruction, stability, or prediction.

The strong claim is mathematical and testable: the system under study admits a generator–configuration structure isomorphic to a free commutative monoid (or a close variant), and there exists an additive cost functional on generator multiplicities such that configuration statistics are reconstructible via a composition operator that factorizes across generators. In that claim, primes are not a decoration; they are the canonical model of irreducible indexing, and the Euler-product thermodynamics is not a metaphor; it is an exact consequence of independence plus additivity.

The boundary between analogy and mathematics is therefore operational:

If prime-level measurements support stable reconstruction of configuration-level statistics across windows and conditions, and if deviations can be systematically attributed to interaction terms, then the framework has crossed from analogy into mathematics. If prime-level signatures are aesthetically pleasing but do not reconstruct, are not stable, or do not beat null models, then the framework is still analogy.

This boundary is crucial because it avoids a common failure mode: admiring prime structure in data without earning the right to interpret it as irreducible causality.

10.2 Dependence on factorization uniqueness and domain encoding

Everything in this paper leans on a single deep property: uniqueness of decomposition into irreducibles. For integers, that property is guaranteed. For real domains, it is not.

Two separate requirements must be satisfied.

First, the domain must have a notion of composition that is stable and meaningful. If objects do not combine in a way that can be modeled as multiplication (or as an operation with similar decomposition behavior), the prime framework is not appropriate.

Second, the encoding from domain objects to integers must preserve that composition in a faithful way. If the encoding is arbitrary, prime factors are not “real,” they are artifacts. The framework then becomes a kind of numerology: it will always find patterns because factorization always exists, but those patterns will not reflect domain structure.

This creates a genuine limitation and a genuine research opportunity.

The limitation is that the method is only as good as the encoding. Encoding design is a major part of the scientific work, not a preprocessing detail.

The opportunity is that encoding faithfulness can be tested. The framework itself supplies checks: stability of inferred spectra across windows, robustness under reweighting, and reconstruction accuracy. If an encoding is wrong, these tests should fail. Thus the method can serve as an encoding validator.

There is also a subtler dependence: even when decomposition is unique, the choice of irreducibles may not be unique across representational levels. A domain can admit multiple valid decompositions depending on what you treat as primitive. For example, in language you can treat characters, morphemes, words, or semantic frames as primitives. Each choice implies a different generator set. The prime framework does not decide which level is “true.” It provides a disciplined way to test which level yields stable generator spectra and predictive reconstruction for the question being asked.

10.3 Extending beyond primes: irreducibles in other algebraic structures

“Prime-only” is best understood as a prototype of a more general principle: irreducible-only aggregation in any structure where objects decompose uniquely into irreducibles.

This points to a broad extension program: replace the integers with other algebraic objects and replace primes with irreducibles in those objects.

Examples include:

Unique factorization domains beyond $\mathbb{Z}$. Polynomials over fields factor uniquely into irreducible polynomials. In such a domain, “prime-only” becomes “irreducible-polynomial-only.” This could be useful for systems where composition corresponds naturally to polynomial multiplication or where signals are modeled by polynomial transfer functions.

Free monoids and formal languages. Strings decompose uniquely into characters, but meaningful irreducibles might be grammar productions rather than characters. If the domain supplies a canonical decomposition into production rules or minimal motifs, then irreducible aggregation becomes a grammar-level analog of prime aggregation.

Graph composition and network motifs. If graphs are composed from motifs under a chosen gluing operation, one could define irreducibles as motifs that cannot be decomposed under that operation. Unique decomposition is rare here, which itself is informative: failure of uniqueness indicates that “basis” is context-dependent and that generator ontology may require probabilistic or minimum-description-length definitions rather than exact algebraic ones.

Noncommutative structures. If composition is noncommutative, occupancy vectors no longer suffice and ordering matters. This leads to richer but more complex analogs: factorization into irreducibles may still exist but may not be unique. A research program here would explore when approximate uniqueness (in a probabilistic sense) is enough to support irreducible aggregation.

The main message is that primes are not the only irreducibles. They are simply the cleanest canonical example. The real research frontier is developing an irreducible aggregation calculus in domains with less perfect factorization, where irreducibles may exist but uniqueness is partial, contextual, or statistical.

10.4 Computational questions: estimation, complexity, and scaling in practice

Even if the conceptual structure is sound, practical deployment raises computational issues.

Factor extraction. Full prime factorization of large integers can be expensive. In many applied settings, this is not fatal because the encoding can be constructed from known factors. But if the integers arise “naturally” and are large, factorization becomes a bottleneck. One research direction is to work with partial factorization or with domain-specific irreducibles that can be extracted efficiently.

Spectrum estimation. Learning $e(p)$ from data requires estimating activation rates, multiplicity distributions, and possibly β . This is feasible if enough samples exist and if the encoding is stable, but identifiability can be limited. A key computational question is which aspects of $e(p)$ are learnable with finite data and how uncertainty should be quantified.

Scaling laws in finite windows. Prime-like signatures often manifest as slow divergences or thinning laws that require large dynamic range to observe. In practice, one needs finite-sample corrections, windowing diagnostics, and confidence intervals that do not overstate evidence. Developing these statistical tools is essential if the framework is to be used as more than an interpretive lens.

Interaction detection. Real systems often violate generator independence. Detecting interactions means measuring correlations among irreducibles, fitting correction terms, and deciding whether interactions are sparse (manageable) or dense (destroying the factorized picture). Computationally, this can become high-dimensional: many generators, many possible couplings. A practical research program needs scalable approximations, such as low-rank interaction models, sparse graphical models over generators, or constrained co-occurrence models.

Model comparison. To claim evidence for multiplicative structure, one must compare against additive and random baselines. This requires standardized metrics: predictive performance on held-out data, stability scores across windows, reconstruction error measures, and complexity penalties to avoid overfitting.

All of these computational questions are tractable in principle but need careful engineering and statistical discipline to avoid turning “prime structure” into a Rorschach test.

10.5 A research agenda: measurable predictions and experimental designs

The most productive next step is to define a research agenda that forces the framework into contact with reality. The agenda should prioritize experiments and datasets where multiplicative composition is plausible and where alternative models are credible competitors.

A first track is controlled synthetic experiments. Construct generative processes where the ground truth is known:

Pure multiplicative independent generator models with known $e(p)$.

Multiplicative models with controlled interactions.

Additive generative models.

Random labeling models.

Then test whether the diagnostics recover the correct structure, whether $e(p)$ is identifiable, and how much data is needed. This establishes power and failure modes before touching messy real data.

A second track is benchmark real domains with natural multiplicative structure. Candidate domains include:

Systems built from modular components with known bill-of-materials style compositions.
Event logs with known templates and parameters where irreducibles can be defined.
Network reliability datasets where root causes and cascades are partially labeled.
Compression and coding tasks where description length is operationally meaningful.

The goal is to avoid domains where encoding is arbitrary. Early wins should come from domains with known composition rules.

A third track is encoding methodology. Develop principles for designing multiplicative encodings:

How to map domain primitives to irreducibles in a way that preserves composition.
How to handle ambiguity and non-unique decompositions.
How to incorporate interactions without abandoning the generator picture.

A fourth track is prediction-focused evaluation. The framework should be judged by what it predicts that alternatives do not. Examples of measurable predictions include:

Prime-like thinning in the discovery rate of new irreducibles as scale increases.
Stability of inferred generator spectra across windows and conditions.
Accurate reconstruction of configuration-level distributions from generator-level spectra.
Characteristic response curves under controlled changes in “temperature-like” parameters (e.g., resource constraints, compression strength, or cost penalties).
Quantifiable interaction signatures when independence fails.

Finally, a fifth track is interpretive integration. The prime-generator ontology should not be used to erase emergence. It should be used to separate layers:

Layer 1: irreducible accounting (prime-level novelty, generator spectra).

Layer 2: emergent structure (interaction terms, co-occurrence patterns, narrative or functional synergies).

Layer 3: domain semantics (what the irreducibles mean, what the interactions mean).

A mature research program would treat Layer 1 as the disciplined baseline and Layer 2 as the domain-specific frontier where meaning and function live.

In summary, the limitations of the framework are also its safeguards: it demands faithful encoding, uniqueness or approximate uniqueness of decomposition, stability across windows, and successful reconstruction. These demands prevent it from devolving into metaphor. The open problems are largely about extending irreducible aggregation beyond perfect unique

factorization, building scalable inference tools, and designing experiments where the framework can be cleanly falsified or supported. If those problems are addressed, prime-only summation becomes not just a technique or an analogy, but a general methodology for identifying and measuring irreducible structure in compositional systems.

11. Conclusion: Prime-Only Summation as a Measure of Independent Reality

The core wager of this paper has been that prime-only summation can be something stronger than a technical restriction or a poetic gesture. It can be a disciplined measure of independence. If a system is built from irreducible generators under a multiplicative (or multiplicatively representable) composition rule, then counting configurations confuses recombination with invention. Prime-only aggregation, by contrast, counts what deserves to be counted once: the basis elements that constitute the system's independent degrees of freedom. The resulting claim is not "primes are mystical," but "irreducibles are primary," and primes provide the cleanest canonical model of what that primacy looks like mathematically.

11.1 Summary of claims and what is genuinely new

This paper made three layered claims, each stronger than the last.

First, a structural claim. The positive integers under multiplication form a free commutative monoid generated by primes. Every integer is uniquely specified by its prime-exponent occupancy vector. If one's target quantity is additive on those occupancies, then primes are the correct primitive index set for that quantity. This is the purely mathematical foundation.

Second, a modeling claim. Assigning a generator cost $e(p)$ and extending it additively to integers produces an induced configuration ensemble with an Euler-product partition function. Macroscopic observables such as free energy, expected energy, and response functions become explicit prime sums. In this view, configuration totals are not primitive; they are reconstructible outputs of generator-level spectra. This turns the "primes as basis" slogan into a working calculus.

Third, an empirical claim. In domains where observations can be encoded faithfully into multiplicative configurations, one can define prime-novelty and prime-surprise functionals that measure irreducible novelty rather than combinatorial proliferation. One can then test for multiplicative generative structure by checking stability across windows, robustness under reweighting, and reconstructability of configuration statistics from generator-level fits, while ruling out additive and random-label alternatives with null models.

What is genuinely new here is not any single mathematical component in isolation. Prime sums, Euler products, and zeta-like forms are classical. The novelty is the *structural reinterpretation assembled into a falsifiable methodology*:

Prime-only summation is elevated from “subset of integers” to “primitive aggregation over independent degrees of freedom.”

Configuration-level proliferation is treated explicitly as an emergent multiplicity generated by composition operators, not as evidence of new primitives.

A thermodynamic calculus is proposed as the bridge between generator spectra and configuration statistics, with clear signatures and failure modes.

An information-theoretic novelty functional is constructed to correct the systematic overcounting that occurs when recombination is mistaken for irreducible innovation.

In short, what is new is the reframing of prime-restricted summation as an *ontological counting rule* paired with explicit reconstruction machinery and an empirical diagnostic protocol.

11.2 The irreducible aggregation lens as a unifying viewpoint

The unifying lens that emerged can be stated compactly:

When composition is multiplicative, irreducibles are the basis of independent reality, and prime-only aggregation is the correct primitive sum for any quantity that tracks independent contribution.

This lens cleanly separates three layers that are often conflated.

Layer 1: Generator reality. These are the irreducible degrees of freedom. They are sparse, and their count grows slowly with scale. Their statistics are prime-indexed.

Layer 2: Configuration reality. These are the emergent composites formed by occupying and combining generators. They are abundant, often explosively so, and their proliferation is a predictable consequence of composition. Their statistics are derived from generator spectra via Euler-product-like operators.

Layer 3: Interaction and meaning. This is where emergence actually lives: correlations among generators, constraints, couplings, semantic synergies, causal cascades. These cannot be reduced to mere generator counts, but they also should not be mistaken for generator creation. They are higher-order structure built on top of the irreducible basis.

The irreducible aggregation lens does not deny emergence. It keeps emergence honest by forcing it to sit on top of a disciplined accounting of irreducibles. It prevents the analytic failure mode in which the mere abundance of composites is interpreted as evidence of abundant independent reality.

That separation is the main conceptual payoff. It gives a principled reason to “skip composites” when measuring independence, while still retaining composites as the domain of emergent structure that can be reconstructed, modeled, and studied.

11.3 Final statement of falsifiable signatures and next steps

A framework that cannot be falsified is not a framework; it is a style. The prime-generator ontology proposed here can be tested and rejected. The key falsifiable signatures are:

Stable generator spectra. If one fits prime activation or occupancy statistics to an implied spectrum $e(p)$, the inferred spectrum should be stable across observation windows and under reasonable reweightings. Instability indicates either non-faithful encoding or absence of a stable generator basis.

Reconstruction success. Generator-level fits should predict configuration-level statistics through composition operators. If prime marginals fit but reconstruction fails systematically, the independence or additivity assumptions are wrong.

Factorization diagnostics. Independent-generator models imply near-factorization: weak residual correlations among primes once the spectrum is accounted for. Strong, persistent correlations imply interactions that must be modeled explicitly or else falsify the independence claim.

Prime-thinning fingerprints. Mode counts and novelty curves should show sublinear thinning consistent with sparse generator emergence, rather than linear growth in independent degrees. If distinct irreducibles grow roughly linearly with scale in a domain where multiplicative composition is claimed, the claim is suspect.

Null-model separation. Prime-based signatures must differ meaningfully from label permutation, factor scrambling, and other baselines that preserve superficial distributions while destroying multiplicative structure. If not, the signatures are artifacts.

These are not aesthetic criteria; they are operational tests.

The next steps follow directly:

First, establish controlled synthetic benchmarks where the ground truth is known, to quantify power, sample requirements, and failure modes.

Second, focus on real domains with natural multiplicative composition and defensible encodings, rather than domains where encoding is arbitrary.

Third, develop statistical tools for finite-sample corrections, spectrum identifiability, and interaction detection, so the framework can operate in realistic conditions.

Fourth, broaden the program beyond primes by seeking irreducible decompositions in other algebraic structures and by developing “approximate irreducibility” methods for domains where uniqueness fails.

If these steps succeed, prime-only summation becomes more than a curiosity. It becomes a general discipline: count what is independent, reconstruct what is composite, and treat emergence as higher-order structure rather than as a mirage of combinatorics. In that sense, prime-only summation is offered here as a measure of independent reality: not because primes are sacred, but because irreducibles are the only honest place to start when the world you observe is built by composition.

12. References

12.1 Analytic and probabilistic number theory (primes, prime sums, factorization)

- G. H. Hardy, E. M. Wright. (2008). *An Introduction to the Theory of Numbers*. Oxford University Press.
- Tom M. Apostol. (1976). *Introduction to Analytic Number Theory*. Springer.
- Harold Davenport. (2000). *Multiplicative Number Theory*. Springer.
- Hugh L. Montgomery, Robert C. Vaughan. (2007). *Multiplicative Number Theory I: Classical Theory*. Cambridge University Press.
- Gérald Tenenbaum. (2015). *Introduction to Analytic and Probabilistic Number Theory*. American Mathematical Society.
- Henryk Iwaniec, Emmanuel Kowalski. (2004). *Analytic Number Theory*. American Mathematical Society.
- Halberstam, Heini, Hans-Egon Richert. (1974). *Sieve Methods*. Academic Press.
- Richard Crandall, Carl Pomerance. (2005). *Prime Numbers: A Computational Perspective*. Springer.
- E. C. Titchmarsh. (1986). *The Theory of the Riemann Zeta-Function*. Oxford University Press.
- Aleksandar Ivić. (2003). *The Riemann Zeta-Function: Theory and Applications*. Dover Publications.
- Harold M. Edwards. (1974). *Riemann's Zeta Function*. Academic Press.
- Paul Erdős, Mark Kac. (1940). The Gaussian law of errors in the theory of additive number theoretic functions. *American Journal of Mathematics*, 62, 738–742.
- Mark Kac. (1959). *Statistical Independence in Probability, Analysis and Number Theory*. Wiley.
- Patrick Billingsley. (1969). On the central limit theorem for the prime divisor functions. *American Mathematical Monthly*, 76, 132–139.

12.2 Euler products, zeta partition functions, and “prime thermodynamics”

- Jean-Benoît Bost, Alain Connes. (1995). Hecke algebras, type III factors and phase transitions with spontaneous symmetry breaking in number theory. *Selecta Mathematica (New Series)*.

- Andreas Knauf. (1999). Number theory, dynamical systems and statistical mechanics. *Reviews in Mathematical Physics*, 11(8), 1027–1060.
- Bernard Julia. (1990). Statistical theory of numbers. In *Number Theory and Physics* (Springer, Berlin), p. 276.
- Bernard Julia. (1994). Thermodynamics of primes. *Physica A*, 203(3–4), 425.
- Emilio Elizalde. (1995). *Ten Physical Applications of Spectral Zeta Functions*. Springer.

12.3 Statistical mechanics, phase transitions, and finite-size analogs

- R. K. Pathria, Paul D. Beale. (2011). *Statistical Mechanics*. Elsevier.
- Kerson Huang. (2001). *Introduction to Statistical Physics*. CRC Press.
- David Ruelle. (1969). *Statistical Mechanics: Rigorous Results*. W. A. Benjamin.
- H. Eugene Stanley. (1971). *Introduction to Phase Transitions and Critical Phenomena*. Oxford University Press.
- Nigel Goldenfeld. (1992). *Lectures on Phase Transitions and the Renormalization Group*. Addison-Wesley.

12.4 Information theory, algorithmic complexity, and description length

- Claude E. Shannon. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27, 379–423 and 623–656.
- Thomas M. Cover, Joy A. Thomas. (2006). *Elements of Information Theory*. Wiley.
- David J. C. MacKay. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.
- Ming Li, Paul Vitányi. (2019). *An Introduction to Kolmogorov Complexity and Its Applications*. Springer.
- Andrey N. Kolmogorov. (1965). Three approaches to the quantitative definition of information. *Problems of Information Transmission*.
- Ray Solomonoff. (1964). A formal theory of inductive inference. *Information and Control*, 7, 1–22 and 224–254.
- Jorma Rissanen. (1978). Modeling by shortest data description. *Automatica*, 14(5), 465–471.

- E. T. Jaynes. (1957). Information theory and statistical mechanics. *Physical Review*, 106, 620–630.

12.5 Model selection, null models, and scaling diagnostics

- Hirotugu Akaike. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6), 716–723.
- Gideon Schwarz. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6(2), 461–464.
- Kenneth P. Burnham, David R. Anderson. (2002). *Model Selection and Multimodel Inference*. Springer.
- Bradley Efron, Robert Tibshirani. (1993). *An Introduction to the Bootstrap*. Chapman and Hall.
- Aaron Clauset, Cosma Rohilla Shalizi, Mark E. J. Newman. (2009). Power-law distributions in empirical data. *SIAM Review*, 51(4), 661–703.

12.6 Networks, reliability, and independent failure modes (applied ontology tests)

- Albert-László Barabási. (2016). *Network Science*. Cambridge University Press.
- Reuven Cohen, Shlomo Havlin. (2010). *Complex Networks: Structure, Robustness and Function*. Cambridge University Press.
- Marvin Rausand, Arnljot Høyland. (2004). *System Reliability Theory*. Wiley.

12.7 Generating functions, composition operators, and asymptotics (configuration reconstruction)

- Philippe Flajolet, Robert Sedgewick. (2009). *Analytic Combinatorics*. Cambridge University Press.

The author has published ~ 5,000 AI-assisted papers at:

<https://www.researchgate.net/profile/Douglas-Youvan/research> . Google Scholar has indexed these preprints, and if you want a particular reference, the AI mode of a Google search (Gemini) has efficiently catalogued these preprints.

